

---

## Javaバイブルシリーズ オブジェクト指向プログラミング 改訂版 別冊

---

### 確認問題解答例

注)プログラムについてはこれが唯一の解答というわけではありません。同様の結果になる異なったプログラムがありますので、是非オリジナルを考えてみてください。

株式会社 **SCC**

## 確認問題 01

- Q1 a) 現実世界  
b) モデリング  
c) OOA  
d) 設計工程  
e) OOT  
f) OOP

Q2 データでも手続きでもなく、「モノ」(オブジェクト)を中心に分析／設計／製造／その他 を行うこと。

- Q3
- |          |   |                         |
|----------|---|-------------------------|
| 手続き指向    | • | システムに必要なデータを中心にしたアプローチ法 |
| データ指向    | • | 手続き指向とデータ指向を合わせたアプローチ法  |
| オブジェクト指向 | • | 機能の手順を中心にしたアプローチ法       |

Q4 a. オ b. デ c. 手 d. 手

- Q5 a. インスタンス  
b. スーパークラス  
c. クラス  
d. 差分プログラミング  
e. オブジェクト

Q6 イ

Q7 ア(情報処理技術者試験 H13 NW 午前問題)

[解説]イは継承の効果、ウは差分プログラミングについて、エはポリモルフィズムと呼ばれる機能の効果について記述されている。

- Q8 • プログラム品質の向上  
• プログラム生産性の向上

Q9 完成品である部品を組み合わせることで生産性が向上し、十分なテストを経た高品質の部品を使用することで、プログラムの品質が向上するから。

## 確認問題 02

- Q1** a) クラス  
b) 変数(フィールド)  
c) メソッド
- Q2** • mainメソッドを持つクラス名とファイル名が異なっている  
• mainはmain(String args[])として引数を受け取らなければならない  
• `z = x + y`に;(セミコロン)が必要  
• javaにはprintf()というメソッドはない  
• }が不足している

**Q3**

```
public class Kakunin02Q3{  
  
    public static void main(String[] args){  
        System.out.println("Javaの勉強がんばります！！");  
    }  
}
```

## 確認問題 03

- Q1** a) 基本データ  
b) 参照(オブジェクト)  
c) byte  
d) short  
e) long  
f) double  
g) true(真)  
h) false(偽)
- Q2** a) class  
b) main  
c) x-y

- Q3**
- |         |        |                              |
|---------|--------|------------------------------|
| byte    | •————• | -128 ~ +127の整数               |
| float   | •      | 16ビットを2の補数表現した整数             |
| char    | •      | 倍精度の浮動小数点数                   |
| long    | •      | -2147483648 ~ +2147483647の整数 |
| boolean | •      | 単精度の浮動小数点数                   |
| double  | •      | Unicodeで表現された1文字             |
| int     | •      | 64ビットを2の補数表現した整数             |
| short   | •      | 真と偽の2種類の状態のみを保持              |

**Q4** 8行目

[解説]int型をshort型に代入しようとしているため精度のエラーが発生する

**Q5**

```
public class Kakunin03Q5{
    public static void main( String[ ] args ){
        double x;
        float y;

        x = 12.5;
        y = (float)x;

        System.out.println( x );
        System.out.println( y );
    }
}
```

## 確認問題 04

**Q1**

```
public class Kakunin04Q1{
    public static void main( String[] args ){
        char a;
        String str;
        str = new String( );

        a = 'ね';
        str = "言ったとおりでしょ!!";

        System.out.println( a );
        System.out.println( str );
    }
}
```

**Q2**

```
public class Kakunin04Q2{
    public static void main(String[] args){
        String str = "こんにちは";
        int x;
        x = str.length();
        System.out.println(str + "の文字数は" + x + "文字です");
    }
}
```

## 確認問題 05

**Q1**

```
public class Kakunin05Q1{
    public static void main(String[] args){
        int[] tbl;
        tbl = new int[3];

        tbl[0] = 5;
        tbl[1] = 8;
        tbl[2] = tbl[0] + tbl[1];
        System.out.println(tbl[2]);
    }
}
```

**Q2**

```
class Kakunin05Q2{
    public static void main( String[] args ){
        int[] tbl;
        int i;
        tbl = new int[4];

        tbl[0] = 100;
        tbl[1] = 200;
        tbl[2] = tbl[0] + tbl[1];
        tbl[3] = tbl[2] * 2;

        for(i=0; i<4; i++){
            System.out.println(tbl[i] );
        }
    }
}
```

**Q3**

```
public class Kakunin05Q3{
    public static void main(String[] args){
        int[] tbl;
        int youso;
        int i;
        tbl = new int[10];

        youso = tbl.length;
        System.out.println(youso);
    }
}
```

**Q4** “null”は、0や空白とは異なり「何も入っていない状態」のことである。

## 確認問題 06

**Q1**

```
public class Kakunin06Q1{
    public static void main( String[] args ){
        int x=5, y=10, z;
        z = x;
        x = y;
        y = z;

        System.out.println( "入れ替え後のx："+x );
        System.out.println( "入れ替え後のy："+y );
    }
}
```

**Q2** 10

9

8

10

9

9

8

**Q3**  $c = a + b / 2;$  を  $c = (a + b) / 2;$ に修正

**Q4** ☐

☐

×(x,yどちらかが正でどちらかが負だと0未満になりfalse)

☐

**Q5**

```
public class Kakunin06Q5{
    public static void main(String[] args){
        int x = -64;
        System.out.println( "xの4倍 : "+(x<<2) );
        System.out.println( "xの1/8 : "+(x>>3) );
    }
}
```

**Q6**

```
if(y != 10){
    x = 3;
}
else{
    x = 10;
}
```



## 確認問題 07

**Q1**

```
public class Kakunin07Q1{
    public static void main( String[ ] args ){
        int x;
        x = -30;

        if(x>=0){
            System.out.println( "変数xは正の数です" );
        }else{
            System.out.println( "変数xは負の数です" );
        }
    }
}
```

**Q2**

```
public class Kakunin07Q2{
    public static void main( String[ ] args ){
        int tokuten;
        tokuten = 80;

        if( tokuten >= 80 ){
            System.out.println( "優です" );
        }else if( tokuten >= 65 ){
            System.out.println( "良です" );
        }else if( tokuten >= 50 ){
            System.out.println( "可です" );
        }else{
            System.out.println( "不可です" );
        }
    }
}
```

**Q3**

```
public class Kakunin07Q3{
    public static void main( String[ ] args ){
        int tsuki;
        tsuki = 10;

        switch( tsuki ){
            case 3:
            case 4:
            case 5:
                System.out.println( "春です" );
                break;
            case 6:
            case 7:
            case 8:
                System.out.println( "夏です" );
                break;
            case 9:
            case 10:
            case 11:
                System.out.println( "秋です" );
                break;
            case 12:
            case 1:
            case 2:
                System.out.println( "冬です" );
                break;
            default:
                System.out.println( "範囲外です" );
                break;
        }
    }
}
```

- Q4** a: 初期化处理  
b: 繰り返しを継続する条件  
c: 繰り返し1回ごとの処理  
d: 繰り返しを継続する条件  
e: `i = 0`  
f: `i < 10`  
g: `i++`

**Q5**

```
public class Kakunin07Q5{
    public static void main(String[ ] args){
        int i;
        int gokei = 0;

        for(i = 2 ; i <= 100 ; i += 2 ){
            gokei += i;
        }
        System.out.println("合計は" + gokei + "です。");
    }
}
```

**Q6**

```
public class Kakunin07Q6{
    public static void main(String[ ] args){
        int i;
        int gokei = 0;

        i = 2;
        while( i <= 100){
            gokei += i;
            i += 2;
        }
        System.out.println("合計は" + gokei + "です。");
    }
}
```

**Q7**

```
public class Kakunin07Q7{
    public static void main(String[ ] args){
        int i,j;

        for( i = 1 ; i <= 9 ; i++ ){
            for( j = 1 ; j <= 9 ; j++){
                System.out.print( i * j + "¥t" );
            }
            System.out.println( );
        }
    }
}
```

**Q8** a: i = 1 ; i <= 5 ; i++  
b: j = 1 ; j <= i ; j++  
c: System.out.print( "¥n" )

**Q9** 解答例1(while構文を使用した場合)

```
public class Kakunin07Q9{
    public static void main(String[ ] args){
        int[ ] seiseki ={57,60,89,32,66,77,92,45};
        int max = 0;
        int min = 100;
        int i=0;

        while( i < seiseki.length){
            if( max < seiseki[i] ){
                max = seiseki[i];
            }
            if( min > seiseki[i]){
                min = seiseki[i];
            }
            i++;
        }
        System.out.println("最高点は" + max + "点です。");
        System.out.println("最低点は" + min + "点です。");
    }
}
```

### 解答例2 (for構文を使用した場合)

```
public class Kakunin07Q9{
    public static void main( String[] args ){
        int[] seiseki={57,60,89,32,66,77,92,45};
        int max = 0;
        int min = 100;

        for( int i = 0 ; i < seiseki.length ; i++ ){
            if( max < seiseki[i] ){
                max = seiseki[i];
            }
            if( min > seiseki[i] ){
                min = seiseki[i];
            }
        }
        System.out.println("最高点は" + max + "点です。");
        System.out.println("最低点は" + min + "点です。");
    }
}
```

### 解答例3 (拡張for構文を使用した場合)

```
public class Kakunin07Q9{
    public static void main( String[] args ){
        int[] seiseki={57,60,89,32,66,77,92,45};
        int max = 0;
        int min = 100;

        for( int n : seiseki ){
            if( max < n ){
                max = n;
            }
            if( min > n ){
                min = n;
            }
        }
        System.out.println("最高点は" + max + "点です。");
        System.out.println("最低点は" + min + "点です。");
    }
}
```

## 確認問題 08

**Q1**

```
public class Kakunin08Q1{

    public static void main(String[] args){

        Kakunin08Q1    instA;

        instA = new Kakunin08Q1();

        instA.asaNoAisatsu();
    }

    public void asaNoAisatsu(){

        System.out.println("おはようございます！");
    }
}
```

**Q2**

```
public class Kakunin08Q2{

    public static void main(String[] args){

        Kakunin08Q2    instA,instB,instC;

        instA = new Kakunin08Q2();
        instB = new Kakunin08Q2();
        instC = new Kakunin08Q2();

        instA.asaNoAisatsu();
        instB.hiruNoAisatsu();
        instC.yoruNoAisatsu();
    }

    public void asaNoAisatsu(){

        System.out.println("おはようございます！");
    }
}
```

```

public void hiruNoAisatsu(){

    System.out.println("こんにちは！");
}

public void yoruNoAisatsu(){

    System.out.println("こんばんは！");
}
}

```

## 確認問題 09

**Q1**

```

public class Kakunin09Q1{

    public static void main(String[] args){

        Gohan1    instA;

        instA = new Gohan1();

        instA.asaGohan();
    }
}

```

```

public class Gohan1{

    public void asaGohan(){

        System.out.println("目玉焼きとみそ汁です。");
    }
}

```

※public classで始まるプログラムは別ファイルにします

```
Q2 public class Kakunin09Q2{

    public static void main(String[] args){

        Gohan2    instA,instB,instC;

        instA = new Gohan2();
        instB = new Gohan2();
        instC = new Gohan2();

        instA.asaGohan();
        instB.hiruGohan();
        instC.yoruGohan();

    }
}
```

```
public class Gohan2{

    public void asaGohan(){

        System.out.println("目玉焼きとみそ汁です。");
    }

    public void hiruGohan(){

        System.out.println("とんかつ定食です。");
    }

    public void yoruGohan(){

        System.out.println("焼肉食べ放題です。");
    }
}
```



## 確認問題 10

**Q1**

```
public class Kakunin10Q1{

    public static void main(String[] args){

        int          kekka;
        int          suchi = 100;
        Kakunin10Q1  instA;

        instA = new Kakunin10Q1();

        kekka = instA.nibai(suchi);

        System.out.println("渡した数値[" + suchi + "]の倍数は" + kekka);
    }

    public int nibai(int suchi){

        return (suchi*2);
    }
}
```

**Q2**

```
public class Kakunin10Q2{

    public static void main(String[] args){

        int suchi = 100;
        int kekka;

        Kakunin10Q2  instA;

        instA = new Kakunin10Q2();

        kekka = instA.nibai(suchi);
        System.out.println("渡した数値[" + suchi + "]の倍数は" + kekka);
        kekka = instA.sanbai(suchi);
        System.out.println("渡した数値[" + suchi + "]の3倍の数は" + kekka);
    }
}
```

続く>>>

```

public int nibai(int suchi){

    return (suchi*2);
}

public int sanbai(int suchi){

    return (suchi*3);
}
}

```

**Q3**

```

public class Kakunin10Q3{

    public static void main(String[] args){

        int suchi;
        int kekka;

        Kakunin10Q2  instA;

        instA = new Kakunin10Q2();

        suchi = Integer.parseInt(args[0]);
        kekka = instA.nibai(suchi);
        System.out.println("渡した数値[" + suchi + "]の倍数は" + kekka);
        kekka = instA.sanbai(suchi);
        System.out.println("渡した数値[" + suchi + "]の3倍の数は" + kekka);
    }

    public int nibai(int suchi){

        return (suchi*2);
    }

    public int sanbai(int suchi){

        return (suchi*3);
    }
}

```

## 確認問題 11

Q1

```
public class KeisanStatic{

    public static int tashisan(int su1,int su2){

        return (su1+su2);
    }

    public static int hikisan(int su1,int su2){

        return (su1-su2);
    }
}
```

```
public class KeisanNomal{

    public int tashisan(int su1,int su2){

        return (su1+su2);
    }

    public int hikisan(int su1,int su2){

        return (su1-su2);
    }
}
```

```
public class Kakunin11Q1{

    public static void main(String[] args){

        int          kekka;
        KeisanNomal   instA;

        instA = new KeisanNomal();

        kekka = instA.tashisan(100,50);
        System.out.println("Nomalの加算結果 : " + kekka);
    }
}
```

続く>>>

```

        kekka = KeisanStatic.tashisan(100,50);
        System.out.println("Staticの加算結果：" + kekka);

        kekka = instA.hikisan(100,50);
        System.out.println("Nomalの減算結果：" + kekka);

        kekka = KeisanStatic.hikisan(100,50);
        System.out.println("Staticの減算結果：" + kekka);
    }
}

```

**Q2**

```

public class Kakunin11Q2{

    public static void main(String[] args){

        int            kekka,su1,su2;
        KeisanNomal    instA;

        instA = new KeisanNomal();

        su1 = Integer.parseInt(args[0]);
        su2 = Integer.parseInt(args[1]);

        kekka = instA.tashisan(su1,su2);
        System.out.println("Nomalの加算結果：" + kekka);

        kekka = KeisanStatic.tashisan(su1,su2);
        System.out.println("Staticの加算結果：" + kekka);

        kekka = instA.hikisan(su1,su2);
        System.out.println("Nomalの減算結果：" + kekka);

        kekka = KeisanStatic.hikisan(su1,su2);
        System.out.println("Staticの減算結果：" + kekka);
    }
}

```

## 確認問題 12

Q1

```
public class HensuCheck{

    public static    int    classHensu=0;
    public          int    instanceHensu=0;

    public void plus(){

        classHensu++;
        instanceHensu++;
    }

    public void minus(){

        classHensu--;
        instanceHensu--;
    }
}
```

```
public class Kakunin12Q1{

    public static void main(String[] args){

        HensuCheck  instA,instB;
        int         i;

        instA = new HensuCheck();
        instB = new HensuCheck();

        for(i=0;i<50;i++){
            instA.plus();
            instB.minus();
        }

        System.out.println("プラス側のクラス変数          : "+instA.classHensu);
        System.out.println("プラス側のインスタンス変数     : "+instA.instanceHensu);
        System.out.println("マイナス側のクラス変数          : "+instB.classHensu);
        System.out.println("マイナス側のインスタンス変数 : "+instB.instanceHensu);
    }
}
```

```
Q2 public class Kakunin12Q2{

    public int bunpu[]={0,0,0,0,0,0,0,0,0,0};

    public static void main(String[] args){

        Kakunin12Q2 instA,instB;
        int i;

        instA = new Kakunin12Q2();
        instB = new Kakunin12Q2();

        instA.ransu();
        instB.ransu();

        for(i=0;i<10;i++){
            System.out.print "[" + i + "]"が"出た回数: ";
            System.out.println(instA.bunpu[i]+instB.bunpu[i]);
        }
    }

    public void ransu(){

        int ret,i;

        for(i=0;i<10;i++){
            ret = (int)(Math.random() * 10);
            bunpu[ret]++;
        }
    }
}
```

## 確認問題 13

Q1

```
public class GamePlayer{

    public void play(String name){

        System.out.println(name + "で遊びました");
    }

    public void play(String name,int jikan){

        System.out.println(name + "を" + jikan + "時間鑑賞しました");
    }
}
```

```
public class Kakunin13Q1{

    public static void main(String[] args){

        GamePlayer    instA;

        instA = new GamePlayer();

        instA.play("格闘ゲーム");
        instA.play("カンフー映画",2);

    }
}
```

**Q2**

```
public class Keisan{

    public int enzan(int su1){

        return (su1*sul1);
    }

    public int enzan(int su1,int su2){

        return (su1 - su2);
    }

    public int enzan(int su1,int su2,int su3){

        return (su1 + su2 + su3);
    }
}
```

```
public class Kakunin13Q2{

    public static void main(String[] args){

        Keisan instA;

        instA = new Keisan();

        System.out.println("数値を1つ[5]渡した結果      : " + instA.enzan(5));
        System.out.println("数値を2つ[8,5]渡した結果    : " + instA.enzan(8,5));
        System.out.println("数値を3つ[11,8,5]渡した結果 : " + instA.enzan(11,8,5));
    }
}
```



## 確認問題 14

Q1

```
public class Car{

    public void drive(int suchi){

        System.out.println( suchi + "Km走りました");
    }
}
```

```
public class Bus extends Car{

    public void noseru(int su){

        System.out.println( su + "人乗せました");
    }

    public void orosu(int su){

        System.out.println( su + "人降ろしました");
    }
}
```

```
public class PatrolCar extends Car{

    public void siren(){

        System.out.println("サイレンを鳴らしました");
    }
}
```

```
public class Kakunin14Q1{

    public static void main(String[] args){

        Bus        instBus;
        PatrolCar   instPCar;

        instBus = new Bus();
        instPCar = new PatrolCar();
    }
}
```

続く>>>

```

        instBus .noseru(5);
        instBus.drive(20);
        instBus.orosu(3);

        instPCar.siren();
        instPCar.drive(10);
    }
}

```

**Q2**

```

public class Kagenzan{

    int kekka;

    public void tashizan(int su1,int su2){

        kekka = su1 + su2;
    }

    public void hikizan(int su1,int su2){

        kekka = su1 - su2;
    }

    public void hyoji(){

        System.out.println("現在の結果は[" + kekka + "]です");
    }
}

public class Shisokuenzan extends Kagenzan{

    public void kakezan(int su1,int su2){

        kekka = su1 * su2;
    }

    public void warizan(int su1,int su2){

        kekka = su1 / su2;
    }
}

```

続<>>>

```

    }
}
public class Kakunin14Q2{

    public static void main(String[] args){

        Shisokuenzan    instA;

        instA = new Shisokuenzan();

        instA.tashizan(10,5);
        instA.hyoji();

        instA.hikizan(10,5);
        instA.hyoji();

        instA.kakezan(10,5);
        instA.hyoji();

        instA.warizan(10,5);
        instA.hyoji();

    }
}

```

## 確認問題 15

**Q1** `public class Bird{`

```

    public void idou(){
        System.out.println( "飛んで移動します。" );
    }
    public void eat(){
        System.out.println( "エサを食べます。" );
    }
}

```

```

public class Suzume extends Bird{

    public void eat(){
        System.out.println( "昆虫などを食べます。" );
    }
}

```

続く>>>

```

    }
}
public class Penguin extends Bird{

    public void idou(){
        System.out.println( "泳いで移動します。" );
    }
    public void eat(){
        System.out.println( "魚を食べます。" );
    }
}

```

```

public class Kakunin15Q1{

    public static void main(String[] args){

        Suzume    instSuzume;
        Penguin    instPenguin;

        instSuzume = new Suzume();
        instPenguin = new Penguin();

        System.out.println( "スズメは" );
        instSuzume.idou();
        instSuzume.eat();

        System.out.println( "ペンギンは" );
        instPenguin.idou();
        instPenguin.eat();
    }
}

```

**Q2** `public class AdultSuzume extends Suzume{`

```

    public void eat(){
        super.eat();
        System.out.println( "大人になると主食が種子に変化します。" );
    }
}

```

続く>>>

```

public class Kakunin15Q2{

    public static void main(String[] args){

        AdultSuzume    instSuzume;
        instSuzume = new AdultSuzume();

        System.out.println( "ススメは" );
        instSuzume.idou();
        instSuzume.eat();
    }
}

```

## 確認問題 16

**Q1**

```

public abstract class Arbeit{

    int    kingaku;

    public abstract void calc(int hour);

    public void hyouji(){
        System.out.println( kingaku + "円入手しました。" );
    }
}

```

```

public class Convenience extends Arbeit{

    public void calc(int hour){
        System.out.println( "コンビニで" +hour + "時間働きました。" );
        kingaku = hour * 1000;
    }
}

```

```

public class CDshop extends Arbeit{

    public void calc(int hour){
        System.out.println( "CDショップで" +hour + "時間働きました。");
        kingaku = hour * 850;
    }
}

```

続く>>>

```
public class GasStation extends Arbeit{

    public void calc(int hour){
        System.out.println( "スタンドで" + hour + "時間働きました。" );
        kingaku = hour * 1000 + 500;
    }
}
```

```
public class Kakunin16Q1{

    public static void main(String[] args){

        GasStation      instA = new GasStation();
        Convenience      instB = new Convenience();
        CDshop           instC = new CDshop();

        instA.calc(3);
        instA.hyouji();
        instB.calc(4);
        instB.hyouji();
        instC.calc(2);
        instC.hyouji();
    }
}
```

**Q2**

```
public abstract class Student{

    public void study(int hour){
        System.out.println( "自宅で" + hour + "時間勉強しました。" );
    }

    public abstract void work(int hour);
}
```

```
public class Pattern1 extends Student{

    public void work(int hour){
        Convenience inst = new Convenience();
        inst.calc(hour);
        inst.hyouji();
    }
}
```

続く>>>

```
public class Pattern2 extends Student{

    public void work(int hour){
        System.out.println( "お手伝いを" + hour + "時間しました。");
    }
}
```

```
public class Kakunin16Q2{

    public static void main(String[] args){

        Pattern1 tomoko      = new Pattern1();
        Pattern1 tatsuhiro    = new Pattern1();
        Pattern2 nobuyasu     = new Pattern2();

        System.out.println( "tomokoの生活です。" );
        tomoko.work(3);
        tomoko.study(3);
        System.out.println( "tatsuhiroの生活です。" );
        tatsuhiro.work(5);
        tatsuhiro.study(1);
        System.out.println( "nobuyasuの生活です。" );
        nobuyasu.work(2);
        nobuyasu.study(5);
    }
}
```

## 確認問題 17

**Q1**

```
public interface Narabikae{

    public void sort(int[] data);

    public void disp();
}
```

```
public class SelectionSort implements Narabikae{

    int[] data;
```

続く>>>

```

public void sort(int[] data){
    int i,j;    //添え字
    int tmp;    //swap用変数

    //並べ替え（選択法）
    for( i = 0 ; i < data.length - 1 ; i++ ){
        for( j = i ; j < data.length ; j++ ){
            //swap
            if( data[i] > data[j] ){
                tmp = data[i];
                data[i] = data[j];
                data[j] = tmp;
            }
        }
    }

    //フィールドに格納
    this.data = new int[data.length];
    for ( i = 0 ; i < data.length ; i++){
        this.data[i] = data[i];
    }
}

public void disp(){
    for ( int i = 0 ; i < data.length ; i++){
        System.out.print(data[i] + " ");
    }
    System.out.println();
}
}

```

```

public class BubbleSort{

    int[] data;

    public void sort(int[] data){
        int i,j;    //添え字
        int tmp;    //swap用変数

        //並べ替え（隣接変換法）

```

続<>>



```

        for( i = data.length - 1 ; i > 0 ; i-- ){
            for( j = 0 ; j < i ; j++ ){
                //swap
                if( data[j] > data[j+1] ){
                    tmp = data[j];
                    data[j] = data[j+1];
                    data[j+1] = tmp;
                }
            }
        }

        //フィールドに格納
        this.data = new int[data.length];
        for ( i = 0 ; i < data.length ; i++){
            this.data[i] = data[i];
        }
    }

    public void disp(){
        for ( int i = 0 ; i < data.length ; i++){
            System.out.print(data[i] + " ");
        }
        System.out.println();
    }
}

```

```

public class Kakunin17Q1{

    public static void main( String[] args ){

        int[] data = {95,66,39,85,60,28,58,19,18,29,77,33,24};
        SelectionSort    sort1 = new SelectionSort();
        BubbleSort        sort2 = new BubbleSort();

        System.out.println( "選択法" );
        sort1.sort(data);
        sort1.disp();

        System.out.println( "隣接交換法" );
        sort2.sort(data);
        sort2.disp();
    }
}

```

```
    }  
}
```

**Q2** `public class Denwa{`

```
    public String no;    //電話番号  
  
    public void tell(){  
        System.out.println("電話できます。");  
    }  
}  
public interface Ongaku{  
  
    public void listen();  
}
```

`public class Keitai extends Denwa implements Ongaku{`

```
    public String address; //メールアドレス  
  
    public void mail(){  
        System.out.println("メールできます。");  
    }  
  
    public void listen(){  
        System.out.println("音楽が聴けます。");  
    }  
}
```

`public class Kakunin17Q2{`

```
    public static void main(String[] args){  
  
        Keitai inst = new Keitai();  
  
        inst.tell();  
        inst.mail();  
        inst.listen();  
    }  
}
```

## 確認問題 18

```
Q1 public class BirdFeature{

    public void showName(){
        System.out.println("鳥の名前を表示します。");
    }

    public void about(){
        System.out.println("特徴や習性を表示します。");
    }
}

public class Kestrel extends BirdFeature{

    public void showName(){
        System.out.println("タカ目ハヤブサ科チョウゲンボウ");
    }

    public void about(){
        System.out.println("素早く羽ばたいてホバリングを行った後、急降下
        して獲物を捕らえる。");
    }
}

public class Merlin extends BirdFeature{

    public void showName(){
        System.out.println("タカ目ハヤブサ科コチョウゲンボウ");
    }

    public void about(){
        System.out.println("日本でも田園なので見ることができる冬鳥。");
        System.out.println("ハトくらいの大きさ。");
    }
}

public class Dictionary{

    public void showBirdFeature(BirdFeature birdFeature){
        birdFeature.showName();
    }
}
```

続く>>>

```

        birdFeature.about();
    }
}

```

```

public class Kakunin18Q1{

    public static void main(String[] args){
        Dictionary dic  = new Dictionary();
        Kestrel kestrel = new Kestrel();
        Merlin merlin  = new Merlin();

        dic.showBirdFeature(kestrel);
        dic.showBirdFeature(merlin);
    }
}

```

**Q2** public interface Communication{

```

    public void communicate();
}

```

```

public class Tiger implements Communication{

    public void communicate(){
        System.out.println("トラと触れ合うのは危険です。");
        System.out.println("遠くからエサを投げましょう。");
    }
}

```

```

public class Mustang implements Communication{

    public void communicate(){
        System.out.println("野生馬は警戒心が強いです。");
        System.out.println("エサを置いたら遠くで見守りましょう。");
    }
}

```

```

public class Dolphin implements Communication{

    public void communicate(){

```

続く>>>

```

        System.out.println("イルカはひとなつっこいです。");
        System.out.println("頭をなでることができます。");
    }
}

```

```

public class JavaSafari{

    public void challenge(Communication animal){
        animal.communicate();
    }
}

```

```

public class Kakunin18Q2{

    public static void main(String[] args){
        JavaSafari safari = new JavaSafari();
        Tiger tiger      = new Tiger();
        Mustang mustang   = new Mustang();
        Dolphin dolphin   = new Dolphin();

        safari.challenge(tiger);
        safari.challenge(mustang);
        safari.challenge(dolphin);
    }
}

```

## 確認問題 19

**Q1**

```

public class Fruit{

    String name;
    int price;

    public Fruit(String name,int price){
        this.name = name;
        this.price = price;
    }

    public int buy(int cnt){
        return price * cnt;
    }
}

```

続く>>>

```

    }
}

public class Kakunin19Q1{

    public static void main(String[] args){

        int total = 0;
        Fruit f1,f2,f3;

        f1 = new Fruit("りんご",100);
        f2 = new Fruit("みかん",50);
        f3 = new Fruit("ぶどう",200);

        total = f1.buy(3) + f2.buy(6) + f3.buy(2);
        System.out.println( "合計は" + total + "円です。" );
    }
}

```

**Q2**

```

public class Country{

    String name;
    double population;

    public Country(){
        name = "日本";
        population = 1.3;
    }

    public Country(String name,double population){
        this.name = name;
        this.population = population;
    }

    public void show(){
        System.out.println( name + "の人口は" + population + "億人
        です。" );
    }
}

```

続く>>>

```
public class Kakunin19Q2{  
  
    public static void main(String[] args){  
  
        Country  inst1,inst2,inst3;  
  
        inst1 = new Country();  
        inst2 = new Country("中国",13.2);  
        inst3 = new Country("インド",11.0);  
  
        inst1.show();  
        inst2.show();  
        inst3.show();  
    }  
}
```

## 確認問題 20

**Q1**

```
package kaku20;

public class Syumi{

    String hobby;

    public Syumi(String hobby){

        this.hobby = hobby;
    }

    public void syumiHyouji(){

        System.out.println("趣味は[" + hobby + "]です");
    }
}
```

```
package kaku20;

public class Namae{

    String name;

    public Namae(String name){

        this.name = name;
    }

    public void namaeHyouji(){

        System.out.println("名前は[" + name + "]です");
    }
}
```

```
public class Kakunin20Q1{

    public static void main(String args[]){
```

続く>>>



```

        kaku20.Namae    instNamae;
        kaku20.Syumi    instSyumi;

        instNamae = new kaku20.Namae("S C C太郎");
        instSyumi = new kaku20.Syumi("盆栽");

        instNamae.namaeHyouji();
        instSyumi.syumiHyouji();

    }
}

```

**Q2**

```

package syoukai;

public class Syumi{

    String hobby;

    public Syumi(String hobby){

        this.hobby = hobby;
    }

    public void syumiHyouji(){

        System.out.println("趣味は[" + hobby + "]です");
    }
}

```

```

package syoukai;

public class Namae{

    String name;

    public Namae(String name){

        this.name = name;
    }
}

```

```

        public void namaeHyouji(){

            System.out.println("名前は[" + name + "]です");
        }
    }

    public class Kakunin20Q2{

        public static void main(String[] args){

            syoukai.Namae    instNamae;
            syoukai.Syumi    instSyumi;

            instNamae = new syoukai.Namae("S C C太郎");
            instSyumi = new syoukai.Syumi("盆栽");

            instNamae.namaeHyouji();
            instSyumi.syumiHyouji();

        }
    }
}

```

## 確認問題 21

**Q1**

```

import syoukai.Syumi;
import syoukai.Namae;

public class Kakunin21Q1{

    public static void main(String[] args){

        Namae    instNamae;
        Syumi    instSyumi;

        instNamae = new Namae("S C C太郎");
        instSyumi = new Syumi("盆栽");

        instNamae.namaeHyouji();
    }
}

```

続<>>>

```

        instSyumi.syumiHyouji();

    }

}

```

## 確認問題 22

**Q1**

```

public class Syain1{

    public int      syainBangou;
    public String   busyo;
    public String   namae;
    public int      kihonkyu;

    Syain1(int syainBangou,String busyo,String namae,int kihonkyu){

        this.syainBangou = syainBangou;
        this.busyo = busyo;
        this.namae = namae;
        this.kihonkyu = kihonkyu;

    }

}

```

```

public class Kakunin22Q1{

    public static void main(String[] args){

        Syain1  instA;

        instA = new Syain1(1,"営業部","S C C二郎",200000);

        System.out.println("社員番号[" + instA.syainBangou + "]");
        System.out.println("部署名  [" + instA.busyo + "]");
        System.out.println("名前    [" + instA.namae + "]");
        System.out.println("基本給  [" + instA.kihonkyu + "]");

    }

}

```

**Q2**

```
public class Syain2{

    private int    syainBangou;
    private String busyo;
    private String namae;
    private int    kihonkyu;

    Syain2(int syainBangou,String busyo,String namae,int kihonkyu){

        this.syainBangou = syainBangou;
        this.busyo = busyo;
        this.namae = namae;
        this.kihonkyu = kihonkyu;

    }

    public int getSyainBangou(){

        return syainBangou;

    }

    public String getBusyo(){

        return busyo;

    }

    public String getNamae(){

        return namae;

    }

    public int getKihonkyu(){

        return kihonkyu;

    }

}
```

```
public class Kakunin22Q2{

    public static void main(String[] args){
```

**続<>>>**

```

        Syain2 instA;

        instA = new Syain2(1, "営業部", "S C C 二郎", 200000);

        System.out.println("社員番号[" + instA.getSyainBangou() + "]);
        System.out.println("部署名  [" + instA.getBusyo() + "]);
        System.out.println("名前      [" + instA.getNamae() + "]);
        System.out.println("基本給    [" + instA.getKihonkyu() + "]);
    }
}

```

### Q3 4種類

### Q4 private、修飾子なし(デフォルト)、protected、public

- Q5 private**      • 自クラス内からのみアクセス可能。
- 修飾子なし      • 同一パッケージ内からアクセス可能。
- protected      • 同一パッケージ内、および継承して作られたサブクラスからアクセス可能。
- public          • どのクラスからもアクセス可能。

### Q6 private、修飾子なし(デフォルト)、protected、public

### Q7 修飾子なし(デフォルト)、public

- Q8 abstract**      • 抽象クラスとも呼ばれ、このクラスから直接インスタンスを生成することはできない。
- final            • 継承(主としてオーバーライド)できないことを意味する。このクラスをもとにインスタンス化は可能。

### Q9 静的メソッドと呼ばれ、インスタンスの生成を行わずに使用することができるメソッド。

### Q10 静的変数と呼ばれ、インスタンスの生成を行わずに使用することができる。またインスタンスが生成された場合には、そのクラスとすべてのインスタンス間でただひとつの領域として共有される。

### Q11 • ローカル変数

メソッドの内側で定義する。そのメソッド内でのみ有効となる。

#### • インスタンス変数

すべてのメソッドの外側で定義する。各インスタンスごとに確保され、クラス内のすべてのメソッドから参照できる。

#### • クラス変数

すべてのメソッドの外側で定義し、かつstatic修飾された変数。そのクラスおよび生成されたインスタンス全体で共通の領域となる。

## 確認問題 23

**Q1**

```
public class Kakunin23Q1{

    public static void main(String[] args){

        Kakunin23Q1  instA;
        int kekka;

        instA = new Kakunin23Q1();

        kekka = instA.jouzan(Integer.parseInt(args[0]),Integer.parseInt(args[1]));

        System.out.println("数値1 [" + args[0] + "]");
        System.out.println("数値2 [" + args[1] + "]");
        System.out.println("乗算結果[" + kekka + "]");
    }

    private int jouzan(int su1,int su2){

        return (su1*su2);
    }
}
```

**Q2**

```
public class Kakunin23Q2{

    public static void main(String[] args){

        Kakunin23Q2  instA;
        int kekka;

        instA = new Kakunin23Q2();

        kekka = instA.jouzan(Integer.parseInt(args[0]),Integer.parseInt(args[1]));

        System.out.println("数値1 [" + args[0] + "]");
        System.out.println("数値2 [" + args[1] + "]");
        System.out.println("乗算結果[" + kekka + "]");
    }
}
```

続<>>>

```

private int jouzan(int su1,int su2){

    return (su1*su2);
}
}

```

**Q3**

```

public class Kakunin23Q3{

    public static void main(String[] args){

        Kakunin23Q3  instA;
        int kekka=0;

        instA = new Kakunin23Q3();

        try{
            kekka = instA.jouzan(Integer.parseInt(args[0]),Integer.parseInt(args[1]));
        }catch(Exception e){
            System.out.println("数字ではない値が渡されました");
        }

        System.out.println("数値1[" + args[0] + "]);
        System.out.println("数値2[" + args[1] + "]);
        System.out.println("乗算結果[" + kekka + "]);
    }

    private int jouzan(int su1,int su2){

        return (su1*su2);
    }
}

```

## 確認問題 24

**Q1**

```
public class AisatsuThread1 extends Thread{

    AisatsuThread1(String name){
        super(name);
    }

    public void run(){

        int i;

        for(i=0;i<20;i++){

            System.out.println(this.getName() + ":" + i);
        }
    }
}
```

```
public class Kakunin24Q1{

    public static void main(String[] args){

        AisatsuThread1  thread1,thread2;

        thread1 = new AisatsuThread1("おはようThread ");
        thread2 = new AisatsuThread1("こんにちはThread");

        thread1.start();
        thread2.start();

    }
}
```

※ クラスKeyboardInputのソースは本編中を参照のこと



**Q2**

```
public class AisatsuThread2 extends Thread{

    AisatsuThread2(String name){
        super(name);
    }

    public void run(){

        int i;

        for(i=0;i<20;i++){

            System.out.println(this.getName() + "：" + i);
            try{
                Thread.sleep(200);
            }catch(Exception e){

            }

        }
    }
}
```

```
public class Kakunin24Q2{

    public static void main(String[] args){

        AisatsuThread2    thread1,thread2;

        thread1 = new AisatsuThread2("おはようThread ");
        thread2 = new AisatsuThread2("こんにちはThread");

        thread1.start();
        thread2.start();

    }
}
```

**Q3**

```
public class AsaThread extends Thread{

    String moji;

    AsaThread(String moji){
        this.moji = moji;
    }

    public void run(){

        int i;

        for(i=0;i<20;i++){

            System.out.println(moji + ":" + i);
            try{
                Thread.sleep(200);
            }catch(Exception e){

            }

        }

    }

}
```

```
public class HiruThread implements Runnable{

    String moji;

    HiruThread(String moji){
        this.moji = moji;
    }

    public void run(){

        int i;

        for(i=0;i<20;i++){

            System.out.println(moji + ":" + i);
            try{
```

続<>>>

```

        Thread.sleep(200);
    }catch(Exception e){

    }
}
}
}
}

```

```

public class Kakunin24Q3{

    public static void main(String[] args){

        AsaThread    thread1;
        HiruThread    instA;
        Thread        thread2;

        thread1 = new AsaThread("おはようThread ");
        instA = new HiruThread("こんにちはThread");
        thread2 = new Thread(instA);

        thread1.start();
        thread2.start();

    }
}

```

## 確認問題 25

**Q1**

```
import java.util.LinkedList;

public class QueueList<T>{

    LinkedList<T> list;

    QueueList(){

        list = new LinkedList<T>();
    }

    public int enqueue(T youso){

        list.addLast(youso);
        System.out.println("追加した要素：" + youso);
        return list.size();
    }

    public T dequeue(){E ret =list.getFirst();

        System.out.println("取り出した要素：" + ret);
        list.removeFirst();
        return ret;
    }
}
```

```
public class Kakunin25Q1{

    public static void main(String[] args){

        QueueList<Integer> list;

        list = new QueueList<Integer>();

        list.enqueue(1);
        list.enqueue(2);
        list.enqueue(3);
    }
}
```

```

        list.dequeue();
        list.dequeue();
        list.dequeue();
    }
}

```

**Q2**

```

public class Kakunin25Q2{

    public static void main(String[] args){

        QueueList<Integer>    quelist;
        StackList<Integer>    stacklist;
        int                    i;

        quelist = new QueueList<Integer>();
        stacklist = new StackList<Integer>();

        //データの格納（3要素）
        System.out.println("キューにデータを追加します");
        quelist.enqueue(1);
        quelist.enqueue(2);
        quelist.enqueue(3);

        System.out.println("スタックにデータを追加します");
        stacklist.push(1);
        stacklist.push(2);
        stacklist.push(3);

        //データの取り出し
        System.out.println("キューからデータを取り出します");
        for(i=0;i<3;i++){
            quelist.dequeue();
        }

        System.out.println("スタックからデータを取り出します");
        for(i=0;i<3;i++){
            stacklist.pop();
        }
    }
}

```

**Q3**

```
public class KakuninClass1{

    public int atai;

    KakuninClass1(int atai){

        this.atai = atai;
    }
}
```

```
public class KakuninClass2{

    private int atai;

    KakuninClass2(int atai){

        this.atai = atai;
    }

    public int getAtai(){

        return atai;
    }
}
```

```
public class KakuninClass3{

    public void hyouji(){

        System.out.println("確認クラス3です");
    }
}
```

```
import java.util.ArrayList;

public class Kakunin25Q3{

    public static void main(String[] args){

        ArrayList<Object> list;
```

**続<>>>**

```

list = new ArrayList<Object>();

list.add(new KakuninClass1(10));
list.add(new KakuninClass2(20));
list.add(new KakuninClass3());

KakuninClass1 class1 = (KakuninClass1)list.get(0);
System.out.println("確認クラス1のatai：" + class1.atai);
KakuninClass2 class2 = (KakuninClass2)list.get(1);
System.out.println("確認クラス2のatai：" + class2.getAtai());
KakuninClass3 class3 = (KakuninClass3)list.get(2);
class3.hyouji();

}
}

```

## 確認問題 26

**Q1** `import java.io.*;`

```

public class KeyboardInput{
    String buff = null;
    BufferedReader inputKB;

    public String inputKeyboard(){
        inputKB = new BufferedReader(
            new InputStreamReader(System.in));

        try{
            buff = inputKB.readLine();
        }catch(IOException e){
            System.out.println("KBエラー");
        }
        return buff;
    }
}

```

続く>>>

```

public class Kakunin26Q1{

    public static void main(String[] args){

        KeyboardInput    input;
        String            name,kaisu;
        int               i;

        input = new KeyboardInput();

        System.out.print("名前を入力してください->");
        name  = input.inputKeyboard();
        System.out.print("回数を入力してください->");
        kaisu = input.inputKeyboard();

        System.out.println("*** 今から表示します ***");
        for(i=0;i<Integer.parseInt(kaisu);i++){
            System.out.println(name);
        }
        System.out.println("***表示を終了しました***");
    }
}

```

**Q2** import java.io.\*;

```

public class Kakunin26Q2{

    public static void main(String[] args){

        BufferedWriter buffw = null;
        KeyboardInput    input;
        String            name,kaisu;
        int               i;

        try{
            buffw = new BufferedWriter(
                        new FileWriter("kakunin24.txt"));

            input = new KeyboardInput();

```



```

        System.out.print("名前を入力してください->");
        name = input.inputKeyboard();
        System.out.print("回数を入力してください->");
        kaisu = input.inputKeyboard();

        System.out.println("*** 今から表示します ***");
        buffw.write("*** 今から表示します ***");
        buffw.newLine();
        for(i=0;i<Integer.parseInt(kaisu);i++){
            System.out.println(name);
            buffw.write(name);
            buffw.newLine();
        }
        System.out.println("***表示を終了しました***");
        buffw.write("***表示を終了しました***");
        buffw.newLine();
    }catch(IOException e){
        System.out.println("書き込みエラーです");
    }finally{
        try{
            buffw.flush();
            buffw.close();
        }catch(Exception e){
            System.out.println("終了処理エラーです");
        }
    }
}
}
}

```

**Q3**

```
import java.io.*;

public class Kakunin26Q3{

    public static void main(String[] args){

        KeyboardInput    input;
        String    atai;
        int        su1,su2,work,gokei,kaishi,main;

        input = new KeyboardInput();

        try{
            System.out.print("1つ目の数値を入力してください->");
            atai = input.inputKeyboard();
            su1 = Integer.parseInt(atai);
            System.out.print("2つ目の数値を入力してください->");
            atai = input.inputKeyboard();
            su2 = Integer.parseInt(atai);

            if(su1>su2){
                work = su1;
                su1 = su2;
                su2 = work;
            }

            kaishi = su1;
            for(gokei=0;kaishi<=su2;kaishi++){
                gokei += kaishi;
            }
            System.out.println "[" + su1 + "]"から" + "[" + su2 + "]"までの合計：" + gokei );
        }catch(Exception e){
            System.out.println("数値以外が入力されました");
        }
    }
}
```

## 確認問題 27

**Q1**

```
import java.sql.*;
import java.util.ArrayList;

public class DBSelect{

    static final String selectAll = "SELECT * FROM eigo";

    ArrayList<Gakusei> selectAll(){

        ArrayList<Gakusei> listRet = null;
        int          tensuu = 0;
        String        number,name;
        Connection con = null;
        Statement stmt = null;
        ResultSet rslt = null;
        listRet = new ArrayList<Gakusei>();

        try{
            Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
            con = DriverManager.getConnection("jdbc:odbc:TestJDBC");
            stmt = con.createStatement();
            rslt = stmt.executeQuery(selectAll);

            while(rslt.next()){
                number = rslt.getString("bangou");
                name = rslt.getString("simei");
                tensuu = rslt.getInt("tensuu");

                listRet.add(new Gakusei(number,name,tensuu));
            }
        }catch(Exception e){
            e.printStackTrace();
        }finally{
            try{
                stmt.close();
                con.close();
            }
```

続<>>>

```

        }catch(Exception e){
            e.printStackTrace();
        }
    }
    return listRet;
}
}

```

```

public class Gakusei{

    private String bangou;
    private String name;
    private int    tensuu;

    Gakusei(String bangou,String name,int tensuu){
        this.bangou = bangou;
        this.name = name;
        this.tensuu = tensuu;
    }

    public String getBangou(){
        return bangou;
    }

    public String getName(){
        return name;
    }

    public int getTensuu(){
        return tensuu;
    }

}

```

```

import java.util.ArrayList;

public class Kakunin27Q1{

    public static void main(String[] args){

```

続<>>>

```

        ArrayList<Gakusei> kekka = new ArrayList<Gakusei>();
        DBSelect sqlSelect = new DBSelect();

        kekka = sqlSelect.selectAll();

        for(int i = 0;i<kekka.size();i++){
            Gakusei inst;

            inst = kekka.get(i);
            System.out.print(inst.getBangou() + " ");
            System.out.print(inst.getName() + " ");
            System.out.println(inst.getTensuu() + " ");
        }
    }
}

```

**Q2** `import java.sql.*;`

```

public class DBInsert{

    static final String insertData = "INSERT INTO eigo VALUES(?,?,?)";

    public void insert(Gakusei gakusei){

        Connection con = null;
        PreparedStatement stmt = null;
        int rslt;

        try{
            Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
            con = DriverManager.getConnection("jdbc:odbc:TestJDBC");
            stmt = con.prepareStatement(insertData);

            stmt.setString(1,gakusei.getBangou());
            stmt.setString(2,gakusei.getName());
            stmt.setInt(3,gakusei.getTensuu());

```

続<>>>

```

        rslt = stmt.executeUpdate();

        System.out.println("***** 1件データを追加しました *****");
        System.out.println("");
    }catch(Exception e){
        e.printStackTrace();
    }finally{
        try{
            stmt.close();
            con.close();
        }catch(Exception e){
            e.printStackTrace();
        }
    }
}
}
}

```

```

public class GakuseiCreate{

    Gakusei  gakusei;

    public Gakusei gakuseiIn(){

        String  bangou,simei;
        int      tensuu;
        KeyboardInput  instKbin = new KeyboardInput();

        System.out.print("学生番号-->");
        bangou = instKbin.inputKeyboard();
        System.out.print("氏名-->");
        simei = instKbin.inputKeyboard();
        System.out.print("点数-->");
        tensuu = Integer.parseInt(instKbin.inputKeyboard());

        return (new Gakusei(bangou,simei,tensuu));
    }
}

```

続<>>

```

import java.util.ArrayList;

public class Kakunin27Q2{

    public static void main(String[] args){

        ArrayList<Gakusei> kekka = new ArrayList<Gakusei>();
        DBSelect  sqlSelect = new DBSelect();
        DBInsert  sqlInsert = new DBInsert();
        GakuseiCreate instCreate = new GakuseiCreate();
        Gakusei  instGakusei;
        int      i;

        kekka = sqlSelect.selectAll();

        for(Gakusei inst : kekka){

            System.out.print(inst.getBangou() + " ");
            System.out.print(inst.getName() + " ");
            System.out.println(inst.getTensuu() + " ");
        }
        System.out.println("***** 全[" + kekka.size() + "]件 *****");
        System.out.println("");

        instGakusei = instCreate.gakuseiIn();

        sqlInsert.insert(instGakusei);

        kekka = sqlSelect.selectAll();

        for(Gakusei inst : kekka){

            System.out.print(inst.getBangou() + " ");
            System.out.print(inst.getName() + " ");
            System.out.println(inst.getTensuu() + " ");
        }
        System.out.println("***** 全[" + kekka.size() + "]件 *****");
        System.out.println("");
    }
}

```

続<>>>

```
}  
}
```

## 確認問題 28

**Q1**

```
import java.awt.*;  
import java.awt.event.*;  
import javax.swing.*;  
  
public class Kakunin28Q1 extends JFrame implements ItemListener, ActionListener{  
    private FlowLayout    flow;  
    private JTextField    text;  
    private JButton       button;  
    private ButtonGroup   btngrp;  
    private JRadioButton  rbtn1,rbtn2,rbtn3;  
    private String        moji;  
    private Container     cntnr;  
  
    Kakunin28Q1(String title){  
  
        super(title);  
        setBounds(200,200,350,200);  
    }  
  
    public static void main(String[] args){  
  
        Kakunin28Q1  dodai;  
  
        dodai = new Kakunin28Q1("Kakunin28Q1");  
        dodai.junbi();  
        dodai.setVisible(true);  
    }  
  
    void junbi(){  
        cntnr = getContentPane();  
        flow = new FlowLayout();  
        cntnr.setLayout(flow);  
  
        btngrp = new ButtonGroup();
```

続<>>>



```

        rbtn1 = new JRadioButton("和食", false);
        rbtn2 = new JRadioButton("フレンチ", false);
        rbtn3 = new JRadioButton("イタリアン", false);
        text = new JTextField(20);
        button = new JButton("今選択されているのは?");

        btngrp.add(rbtn1);
        btngrp.add(rbtn2);
        btngrp.add(rbtn3);
        cntnr.add(rbtn1);
        cntnr.add(rbtn2);
        cntnr.add(rbtn3);
        cntnr.add(button);
        cntnr.add(text);

        setDefaultCloseOperation(EXIT_ON_CLOSE);
        rbtn1.addItemListener(this);
        rbtn2.addItemListener(this);
        rbtn3.addItemListener(this);
        button.addActionListener(this);
    }

    public void itemStateChanged(ItemEvent e){
        moji = ((JRadioButton)e.getItemSelectable()).getText();
    }

    public void actionPerformed(ActionEvent e){
        if(moji != null){
            text.setText("よし！" + moji + "にしよう！");
        }else{
            text.setText("う〜ん 何にしよう？");
        }
    }
}

```

## 確認問題 29

**Q1**

```
import java.applet.Applet;
import java.awt.Graphics;
import java.awt.Button;
import java.awt.Choice;
import java.awt.Color;
import java.awt.event.ActionListener;
import java.awt.event.ItemListener;
import java.awt.event.ActionEvent;
import java.awt.event.ItemEvent;

public class Kakunin29Q1 extends Applet implements ItemListener,ActionListener{

    private Choice  houseChoice;
    private Button  kenchiku;
    private Button  clear;
    private String[] itemHouse = {"煙突","屋根","壁","窓"};
    private int x[] = { 175,75,275 };
    private int y[] = { 90,200,200 };
    private int index=-1;
    private boolean clearFlg = false;

    public void init(){
        int i;

        houseChoice = new Choice();
        kenchiku = new Button("建築");
        clear = new Button("クリア");

        for(i=0;i<4;i++){
            houseChoice.addItem(itemHouse[i]);
        }

        add(houseChoice);
        add(kenchiku);
        add(clear);
        kenchiku.addActionListener(this);
        clear.addActionListener(this);
```

続<>>>

```

        houseChoice.addItemListener(this);
    }

    public void paint(Graphics g){

        if(clearFlg == true){
            g.setColor(getBackground());
            g.fillRect(0,0,getSize().width,getSize().height);
            clearFlg = false;
        }else{
            switch(index){

                case 0:                                //煙突
                    g.fillRect(220,130,20,50);
                    break;

                case 1:                                //屋根
                    g.setColor(Color.red);
                    g.fillPolygon(x,y,3);
                    break;

                case 2:                                //壁
                    g.setColor(Color.blue);
                    g.fillRect(100,200,150,100);
                    break;

                case 3:                                //窓
                    g.setColor(Color.white);
                    g.fillRect(150,225,50,40);
                    g.setColor(Color.black);
                    g.drawLine(175,225,175,265);
                    g.drawLine(150,245,200,245);
                    break;

                default:
                    break;

            }
        }
    }

    public void update(Graphics g){
        paint(g);
    }

```

続<>>>

```

public void actionPerformed(ActionEvent e){
    if(e.getSource() == kenchiku){
        if(index == -1){
            index = 0;
        }
    }else{
        clearFlg = true;
    }
    repaint();
}

public void itemStateChanged(ItemEvent e){
    index = ((Choice)e.getSource()).getSelectedIndex();
}
}

```

```

<HTML>

<applet code="Kakunin29Q1.class" width=500 height=400>
</applet>

</HTML>

```