

=====

Created by YUEN Yiu Yeung, 24-03-2021

E: Circle.java

Class Circle - Object-oriented Programming(abstract class and methods)

@YUEN Yiu Yeung

@2021-01-31

public class Circle extends Shape

private double radius;

private Point center;

```
public Circle(double r, double x, double y) {
    super("Circle");
    setRadius(r);
    setCenter(new Point (x,y));
}
```

public double getRadius() {return radius;}

public Point getCenter() {return center;}

```
public void setRadius(double r) {
    this.radius = r;
}
```

```
public void setCenter(Point center) {
```

enter input while your programming is running

REFERENCE :

```
[ 3 8 13 ]
3 ]
00 ]
[ 3 ]
]
11 16 20 25 ]
ng : [ 3 ]
8 ]
[ 33 ]
5 ]
5 ]
]
]
[ 11 20 25 26 26 34 ]
[ 29 ]
: [ 20 ]
: [ 19 34 ]
[ 3 ]
: [ 3 ]
5 22 23 ]
[ 10 19 23 30 30 ]
: [ 16 25 ]
: [ 15 22 ]
[ 33 34 ]
6 ]
6 ]
```

ally terminated.

=====

Created by YUEN Yiu Yeung, 24-03-2021

E: Coffee.java

Class Coffee - Object-oriented Programming(abstract class and methods)

@YUEN Yiu Yeung

@2021-01-31

```
public class Coffee extends Drink {
    protected boolean isSweet;
```

```
    public Coffee(String name, int price, int volume, boolean isSweet) {
        super(name, price, volume);
        this.isSweet = isSweet;
    }
```

```
    public String toString() {
        if (isSweet)
            return super.toString() + ", sweet";
        else
            return super.toString() + ", not sweet";
    }
```

```
    public void setPrice(int price) {
        if (price < 10)
            this.price = 10;
```

enter input while your programming is running

```

    if (price < 10)
        this.price = 10;
    else
        super.setPrice(price);
}

```

ERENCE:

```

3 ]
[ 3 8 11 ]
8 ]
[ 3 ]
ng : [ 3 ]
[ 11 16 ]
5 ]
5 ]
]
]
[ 9 11 13 13 17 ]
[ 3 ]
11 12 ]
: [ 3 ]
11 12 23 24 25 27 ]
: [ 23 27 ]
: [ 16 18 20 ]
[ 11 12 ]

```

ally terminated.

=====

Created by YUEN Yiu Yeung, 24-03-2021

E: Drink.java

Class Drink - Object-oriented Programming (abstract class and methods)

@YUEN Yiu Yeung

@2021-01-31

```

public class Drink extends Food {
    protected int volume;

    public Drink (String name, int price, int volume) {
        super(name, price);
        this.volume = volume;
    }

    public void setPrice(int price) {
        if (price < 5)
            this.price = 5;
        else
            this.price = price;
    }

    public String toString() {
        return super.toString() + ", volume=" + volume;
    }
}

```

enter input while your programming is running

```

public String toString() {
    return super.toString() + ", volume=" + volume;
}

```

REFERENCE:

```

2 ]
2 7 10 ]
2 ]
2 ]
g : [ 2 ]
10 22 ]
4 ]
]
]
[ 2 ]
0 11 ]
[ 2 ]
10 11 15 16 17 19 19 ]
[ 15 ]
[ 22 23 ]
8 10 12 12 23 ]

```

lly terminated.

=====

Created by YUEN Yiu Yeung, 24-03-2021

E: Food.java

Abstract Class Food - Object-oriented Programming (abstract class and methods)

@YUEN Yiu Yeung

@2021-01-31

```

public abstract class Food {
    protected String name;
    protected int price;

    public Food() {
        name = null;
        price = 0;
    }

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

    public String toString() {
        return "name=" + name + ", price=" + price;
    }
}

```

enter input while your programming is running



```

public String toString() {
    return "name=" + name + ", price=" + price;
}

public abstract void setPrice(int price);

```

REFERENCE:

```

[ 3 ]
3 ]
8 8 12 17 ]
[ 3 ]
ng : [ 3 ]
[ 9 17 22 ]
5 ]
5 ]
]
]
[ 3 ]
9 13 17 18 18 23 ]
[ 3 ]
10 14 17 19 23 26 ]
[ 19 26 ]
[ 22 ]

```

lly terminated.



=====

ated by YUEN Yiu Yeung, 24-03-2021

: PartTimeStaff.java

class TestStaff - Object-oriented Programming(Interface)

YUEN Yiu Yeung

2021-01-31

ic class PartTimeStaff extends Staff implements Salary {

private int workingHour;

public PartTimeStaff(String name, int id, char grade, int workingHour) {

super(name, id, grade);

this.workingHour = workingHour;

}

public void display() {

System.out.println("Name: " + this.name + "; ID: " + this.id + "; Grade: " + this.grade + "; Working Ho

this.workingHour + "; Salary: " + computeSalary());

}

public int computeSalary() {

switch (grade) {

case 'A': return SALARY\_A;

case 'B': return SALARY\_B;

case 'C': return SALARY\_C;

ERENCE :

```

3 ]
: [ 3 ]
[ 3 ]
taff : [ 8 11 ]
ng : [ 3 ]
: [ 23 ]
: [ 24 ]
: [ 25 ]
HER : [ 26 ]
[ 8 ]
8 ]
[ 11 ]
[ 17 ]
: [ 3 ]
5 ]
5 ]
]
lary : [ 18 21 ]
[ 16 ]
11 12 17 22 ]
12 17 ]
11 12 17 ]
: [ 3 ]
7 ]
[ 17 ]
ur : [ 9 11 13 13 18 ]

```

ally terminated.

=====

Created by YUEN Yiu Yeung, 24-03-2021

FILE: Salary.java

\*  
Class TestStaff - Object-oriented Programming(Interface)

@YUEN Yiu Yeung

@2021-01-31

```

/
public interface Salary {
    public final int SALARY_A = 4000;
    public final int SALARY_B = 3000;
    public final int SALARY_C = 2000;
    public final int SALARY_OTHER = 1000;

    public abstract int computeSalary();
}

```

REFERENCE:

```

3 ]
: [ 3 ]
[ 3 ]
ng : [ 3 ]
: [ 9 ]
: [ 10 ]
: [ 11 ]

```

enter input while your programming is running

```
public interface Salary {
    public final int SALARY_A = 4000;
    public final int SALARY_B = 3000;
    public final int SALARY_C = 2000;
    public final int SALARY_OTHER = 1000;

    public abstract int computeSalary();
}
```

REFERENCE:

```
3 ]
: [ 3 ]
[ 3 ]
ng : [ 3 ]
: [ 9 ]
: [ 10 ]
: [ 11 ]
HER : [ 12 ]
[ 8 ]
: [ 3 ]
5 ]
5 ]
]
Salary : [ 14 ]
: [ 3 ]
```

ally terminated.

```
=====
```

eated by YUEN Yiu Yeung, 24-03-2021

E: Staff.java

Class TestStaff - Object-oriented Programming(Interface)

@YUEN Yiu Yeung

@2021-01-31

```

public abstract class Staff{
    protected String name;
    protected int id;
    protected char grade;

    public Staff(String name, int id, char grade) {
        this.name = name;
        this.id = id;
        this.grade = grade;
    }

    public abstract void display();

```

RENCE:

3 ]

: [ 3 ]

2 1

enter input while your programming is running

protected char grade;

```
public Staff(String name, int id, char grade) {
    this.name = name;
    this.id = id;
    this.grade = grade;
}
public abstract void display();
```

REFERENCE:

```
3 ]
: [ 3 ]
3 ]
g : [ 3 ]
8 13 ]
9 13 ]
: [ 3 ]
5 ]
5 ]
]
[ 18 ]
11 13 16 16 ]
13 15 15 ]
13 14 14 ]
[ 3 ]
```

lly terminated.