

## Exercise 1: Internationalization

### Question 1:

Xem lại **Exercise 3: Date Format** Assignment 2

## Exercise 2: Annotations

### Question 1: @Deprecated

Khai báo 1 instance có datatype Date và có value là ngày 18/05/2020.

Bạn sẽ thấy java cảnh báo, hãy tắt cảnh báo này đi

```
public class Exercise2 {  
    public void question1() {  
        Date date1 = new Date();  
        System.out.println(date1);  
    }  
}
```

### Question 2: @Deprecated

Tạo 1 class student id, name với id là int và tự động tăng

B1: Sau đó tạo getter cho id

→ Tạo Class Student:

```
package com.vti.entity;  
  
public class Student {  
    public static int COUNT = 0;  
    private int id;  
    private String name;  
  
    public Student(String name) {  
        super();  
        this.id = ++COUNT;  
        this.name = name;  
    }  
  
    /**  
     * @return the id  
     */  
    @Deprecated  
    public int getId() {  
        return id;  
    }  
  
    public String getIdWithMSV() {  
        return "MSV" + id;  
    }  
  
    /**  
     * @return the name  
     */  
    public String getName() {  
        return name;  
    }  
  
    @Override  
    public String toString() {  
        return "Student [id=" + id + ", name=" + name + "]";  
    }  
}
```

```
}
```

B2: Tại main() method, ta sẽ tạo instance của student có name = "Nguyễn Văn A" và in ra tên của student.

B3: Sau đó hệ thống thay đổi method getter của id, bạn sẽ phải tạo ra getter cho id như sau (viết thêm method sau vào)

Method sẽ return ra String là "MSV: " + id (nghĩa là thêm chữ MSV vào trước id

```
@Deprecated
    public int getId() {
        return id;
    }

    public String getIdWithMSV() {
        return "MSV" + id;
    }
```

B4: bạn sẽ đánh dấu method default getter (ở bước 1) là cũ và tạo comment để những người sau sẽ sử dụng method mới (là method ở bước 3)

B5: tại method main() viết thêm demo sử dụng method getter mới này (không xóa code cũ đi)

```
public void question2() {
    System.out.println("Tạo sinh viên với phương thức getID ban đầu:");
    Student stOld = new Student("StudentOld");
    System.out.println("Sinh viên tạo là ID: " + stOld.getId() + " Name: " + stOld.getName());

    System.out.println("Sử dụng phương thức getID bản mới cập nhật getIdWithMSV(): ");
    System.out.println("Sinh viên tạo là ID: " + stOld.getIdWithMSV() + " Name: " + stOld.getName());
}
```

### Exercise 3: Inner class

#### Question 1:

Tạo class CPU có property: price

Tạo 2 inner class Processor, Ram

Với Processor có các property: coreAmount, manufacturer và method getCache() và return ra 4.3

Với Ram có các property: memory, manufacturer và method getClockSpeed() và return ra 5.5

Sau đó khởi tạo object CPU và in ra cache của Processor và clockSpeed của Ram

→ Tạo class CPU: Class CPU chỉ để thuộc tính Price, mà không cần để thuộc tính của Processor và Ram, 2 thuộc tính của Processor và Ram sẽ thể hiện trong các Inner Class.

```
package com.vti.entity;

public class CPU {
    private float price;

    public CPU(float price) {
        super();
        this.price = price;
    }

    public class Processor {
        private int coreAmount;
        private String manufacturer;

        public Processor(int coreAmount, String manufacturer) {
            super();
            this.coreAmount = coreAmount;
            this.manufacturer = manufacturer;
        }

        public Double getCache() {
            return 4.3;
        }
    }

    public class Ram {
        private int memory;
        private String manufacturer;

        public Ram(int memory, String manufacturer) {
            super();
            this.memory = memory;
            this.manufacturer = manufacturer;
        }

        public double getClockSpeed() {
            return 5.5;
        }
    }
}
```

### → Tạo Class Demo:

Với chương trình này khi khởi tạo đối tượng, cần khởi tạo ở class ngoài cùng trước CPU, sau đó mới khởi tạo ở các Class bên trong:

```
CPU cpu1 = new CPU(10);
CPU.Processor pro1 = cpu1.new Processor(4, "Intel");
CPU.Ram ram1 = cpu1.new Ram(16, "Kingston");
```

```
public void question1() {
    CPU cpu1 = new CPU(10);
    CPU.Processor pro1 = cpu1.new Processor(4, "Intel");
    CPU.Ram ram1 = cpu1.new Ram(16, "Kingston");
    System.out.println("Khởi tạo CPU -->OK");

    System.out.println("In thông tin Processor: ");
    System.out.println("Cache Processor: " + pro1.getCache());
    System.out.println("Clock Speed Ram: " + ram1.getClockSpeed());
}
```

Khi lấy các thuộc tính, thì thuộc tính của class nào thì lấy Instance của lớp đó ra lấy:

```
System.out.println("Cache Processor: " + pro1.getCache());
System.out.println("Clock Speed Ram: " + ram1.getClockSpeed());
```

### → output:

```
1
Khởi tạo CPU -->OK
In thông tin Processor:
Cache Processor: 4.3
Clock Speed Ram: 5.5
---          Chọn Question:          ---
---          1. Question 1           ---
---          2. Question 2           ---
---          3. Exit                  ---
```

### Question 2:

Tạo Car có property: name, type

Tạo inner class Engine có property engineType và tạo getter, setter cho property engineType

Khởi tạo object Car có name = Mazda, type = 8WD, có loại động cơ là "Crysler".

Sau đó in ra thông tin của động cơ

### → Tạo Class Car, class Engine là inner class.

```
package com.vti.entity;

public class Car {
    private String name;
    private String type;

    public Car(String name, String type) {
        super();
        this.name = name;
        this.type = type;
    }

    @Override
```

```

public String toString() {
    return "Car [name=" + name + ", type=" + type + "]";
}

public class Engine {

    private String engineType;

    public Engine(String engineType) {
        super();
        this.engineType = engineType;
    }

    @Override
    public String toString() {
        return "Engine [engineType=" + engineType + "]";
    }

    /**
     * @return the engineType
     */
    public String getEngineType() {
        return engineType;
    }

    /**
     * @param engineType the engineType to set
     */
    public void setEngineType(String engineType) {
        this.engineType = engineType;
    }

}
}

```

```

public void question2() {
    Car car = new Car("Mazda", "8WWD");
    Car.Engine engine = car.new Engine("Crysler");
    System.out.println("Tạo Car -->OK");
    System.out.println("Thông tin xe: " + car);
    System.out.println("Thông tin động cơ: " + engine);
}

```

**Question 3:** output của main ra gì?

```

1 public class OuterClass {
2
3     public void show() {
4         InnerClass innerClass = new InnerClass();
5         innerClass.show();
6     }
7
8     public class InnerClass {
9
10        public void show() {
11            System.out.println("Đây là inner class");
12        }
13    }
14 }

```

```

1 public class MainClass {
2
3     public static void main(String[] args) {
4         OuterClass outerClass = new OuterClass();
5         outerClass.show();
6
7         InnerClass innerClass = outerClass.new InnerClass();
8         innerClass.show();
9     }
10
11 }

```

```
package com.vti.entity;
```

```

public class OuterClass {
    public void show() {
        InnerClass innerClass = new InnerClass();
        innerClass.show();
    }

    public class InnerClass {

        public void show() {
            System.out.println("Đây là InnerClass");
        }
    }
}

```

```

public void question3() {
    OuterClass outerClass = new OuterClass();
    outerClass.show();
    OuterClass.InnerClass innerClass = outerClass.new InnerClass();
    innerClass.show();
}

```

**Question 4:** output của đoạn sau ra gì?

```

public class NgayThangNam {
    public int ngay, thang, nam;

    public class GioPhutGiay {
        public int gio, phut, giay;

        public void xuấtThôngTin() {
            System.out.println("Ngày: " + ngay + "/" + thang + "/" + nam);
            System.out.println("Giờ: " + gio + ":" + phut + ":" + giay);
        }
    }
}

```

```

1 public class MainClass {
2
3     public static void main(String[] args) {
4         NgayThangNam date = new NgayThangNam();
5         date.ngay = 31;
6         date.thang = 10;
7         date.nam = 2017;
8
9         GioPhutGiay time = date.new GioPhutGiay();
10        time.gio = 10;
11        time.phut = 15;
12        time.giay = 30;
13
14        time.xuấtThôngTin();
15    }
16
17 }

```

```

package com.vti.entity;

public class NgayThangNam {
    public int ngay, thang, nam;

    public class GioPhutGiay {
        public int gio, phut, giay;

        public void xuấtThôngTin() {
            System.out.println("Ngày: " + ngay + " Tháng: " + thang + "
Năm: " + nam);
            System.out.println("Giờ: " + gio + " Phút: " + phut + "
Giây: " + giay);
        }
    }
}

```

```

public void question4() {

```

```
        NgayThangNam date = new NgayThangNam();  
        date.ngay = 31;  
        date.thang = 10;  
        date.nam = 2017;  
  
        GioPhutGiay time = date.new GioPhutGiay();  
        time.gio = 10;  
        time.phut = 15;  
        time.giay = 30;  
        time.xuatThongTin();  
    }
```

Created By DaoNQ - VTI Academy