

Trình độ: ĐẠI HỌC; Ngày thi: 16/6/2023

Môn: LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

(Đáp án - thang điểm gồm 04 trang)

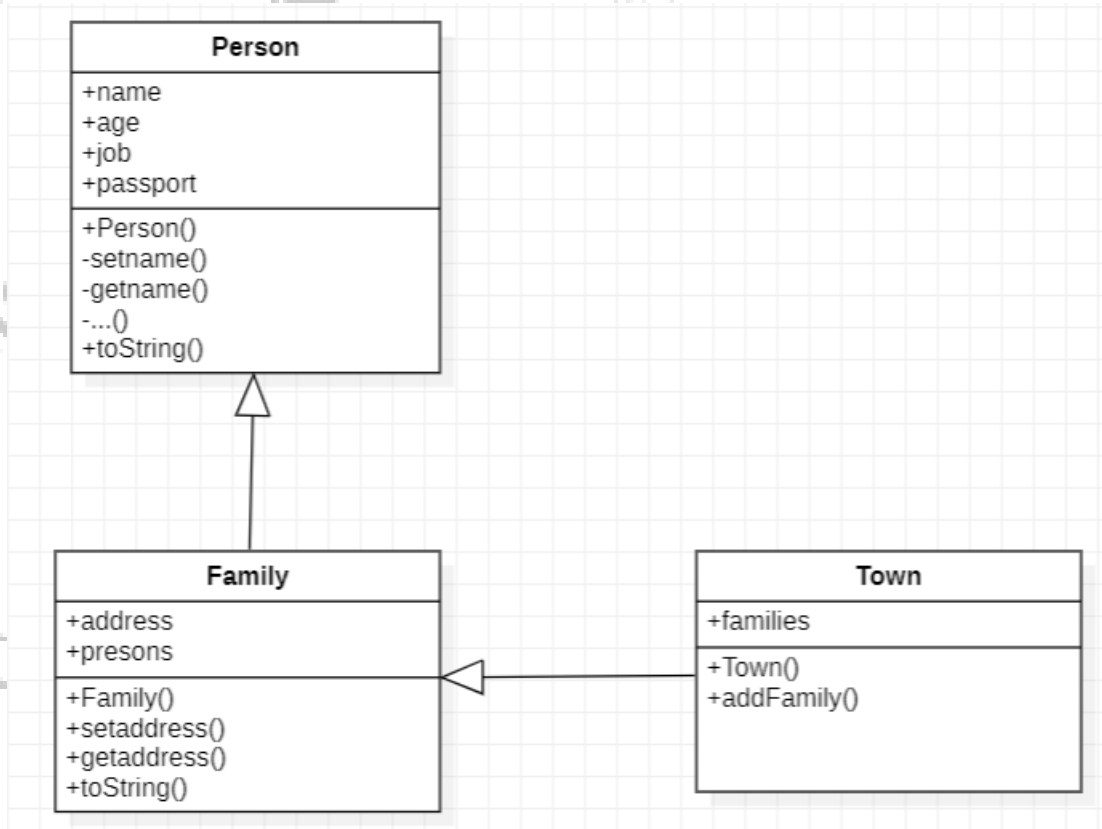
ĐÁP ÁN ĐỀ THI CHÍNH THỨC

Câu	Phần	Nội dung	Thang điểm
I		TRẮC NGHIỆM	5,0
1	C	Chương trình viết bằng Java chỉ có thể được trên môi trường linux	0.25
2	A	2	0.25
3	D	String	0.25
4	A	Tham số đầu tiên của danh sách tham số	0.25
5	D	.class	0.25
6	A	Peter	0.25
7	C	Là một thành phần của Java platform dùng để đọc mã bytecode trong file .class	0.25
8	A	class Cat{}	0.25
9	C	12	0.25
10	B	Không có lỗi biên dịch	0.25
11	A	public class Cat{ String name; public Cat(){ String name = "noname"; } }	0.25
12	C	false	0.25
13	B	Nhập dữ liệu và tính tổng mảng nguyên	0.25
14	D	private class test	0.25
15	A	Lỗi biên dịch	0.25
16	C	Overriding chỉ xảy ra giữa lớp cha và lớp con	0.25
17	B	Là thư viện chứa các thành phần phần mềm tạo sẵn cung cấp các chức năng cho chương trình Java	0.25
18	C	12	0.25
19	B	Nhập một chuỗi	0.25
20	D	Scanner	0.25

II		LẬP TRÌNH	5,0
1	Câu 1	Vẽ sơ đồ UML cho các lớp liên quan.	1.5
2	Câu 2	Xây dựng lớp <i>Ngươi (Person)</i> để quản lý thông tin của mỗi cá nhân	1.5
3	Câu 3	Xây dựng lớp <i>HoGiaDinh (Family)</i> để quản lý thông tin của hộ gia đình	1.0
4	Câu 4	Xây dựng lớp <i>KhuPho (Town)</i> để quản lý các thông tin của hộ gia đình	1.0
		Tổng điểm	10,0 đ

Đáp án (Lập Trình)

1. Vẽ sơ đồ UML cho các lớp liên quan



2. Hãy xây dựng lớp *Ngươi (Person)* để quản lý thông tin của mỗi cá nhân.

```

public class Person {
    private String name;
    private int age;
    private String job;
    private String passport;
    public Person(String name, int age, String job, String passport) {
        this.name = name;
    }
  }
  
```

```

        this.age = age;
        this.job = job;
        this.passport = passport;
    }
    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public int getAge() {
        return age;
    }

    public void setAge(int age) {
        this.age = age;
    }

    public String getJob() {
        return job;
    }
    public void setJob(String job) {
        this.job = job;
    }

    public String getPassport() {
        return passport;
    }
    public void setPassport(String passport) {
        this.passport = passport;
    }

    @Override
    public String toString() {
        return "Person{" +
            "name='" + name + '\'' +
            ", age=" + age +
            ", job='" + job + '\'' +
            ", passport='" + passport + '\'' +
            '}';
    }
}

```

3. Xây dựng lớp *HoGiaDinh (Family)* để quản lý thông tin của từng hộ gia đình

```

import java.util.List;
public class Family {
    private List<Person> presons;

```

```

private String address;
public Family(List<Person> presons, String address) {
    this.presons = presons;
    this.address = address;
}
public List<Person> getPresons() {
    return presons;
}
public void setPresons(List<Person> presons) {
    this.presons = presons;
}
public String getAddress() {
    return address;
}
public void setAddress(String address) {
    this.address = address;
}
@Override
public String toString() {
    return "Family{" +
        "presons=" + presons +
        ", address='" + address + '\'' +
        '}';
}
}

```

4. Xây dựng lớp *KhuPho (Town)* để quản lý các thông tin của từng hộ gia đình

```

import java.util.ArrayList;
import java.util.List;
public class Town {
    List<Family> families;
    public Town() {
        this.families = new ArrayList<>();
    }
    public List<Family> getFamilies() {
        return families;
    }
    public void setFamilies(List<Family> families) {
        this.families = families;
    }
    public void addFamily(Family family) {
        this.families.add(family);
    }
}

```