جامعة عفت

EFFAT UNIVERSITY

CS2132-2

OOP - Spring 2022

Author: **Reem Alsharabi**

ID: **S20106353**

Instructor: **Dr. Fidaa Abed**

Date of Submission: **05/04/2022**

Contents

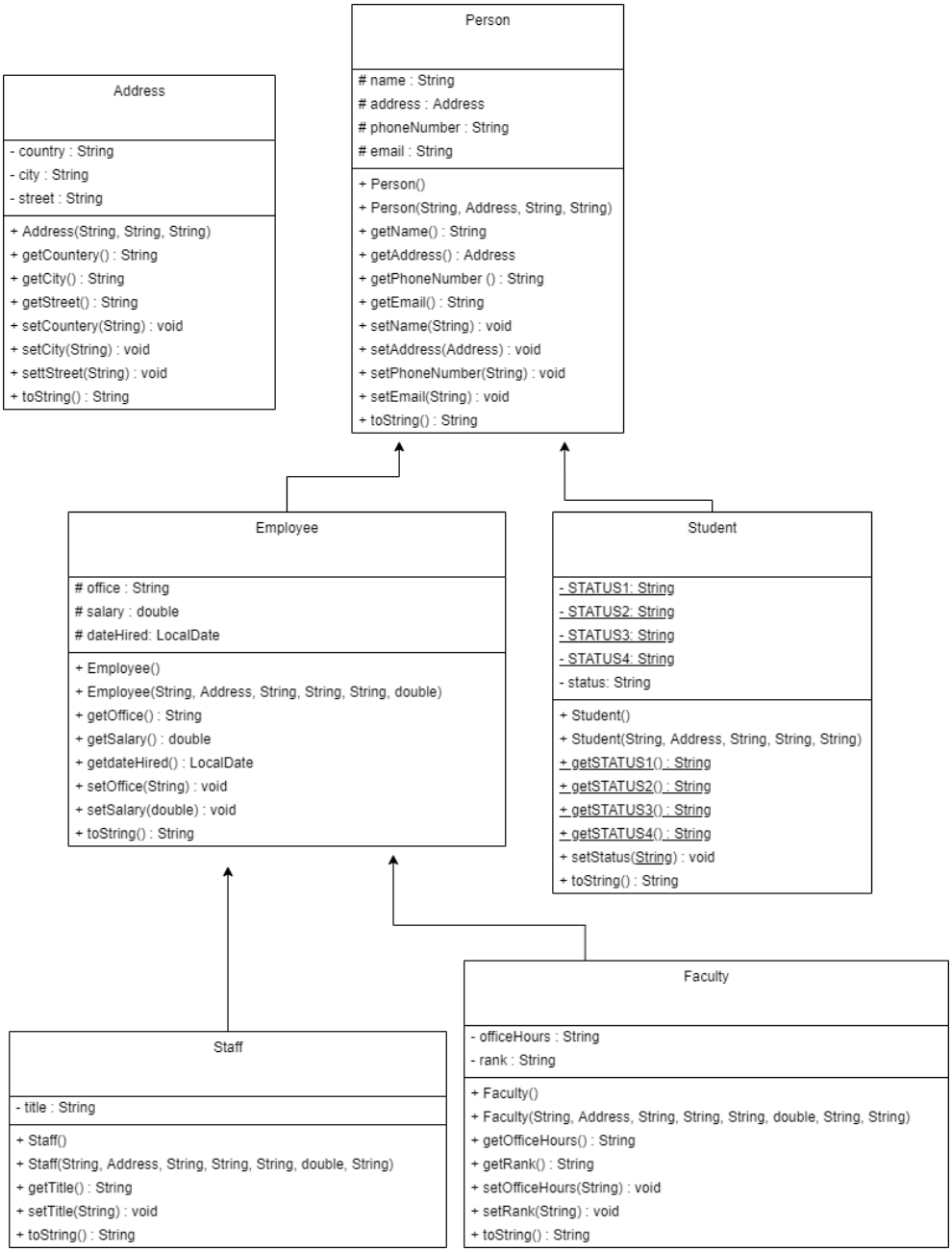
1 Objectives

- Applying the inheritance relationship between classes

2 Questions

2.1 Question 1

- UML



- Person

```
public class Person
{
    protected String name;
    protected Address address;
    protected String phoneNumber;
    protected String email;
    public Person() {}
    public Person(String n, Address a, String p, String e)
    {
        name = n;
        address = a;
        phoneNumber = p;
        email = e;
    }
    public String getName()
    {
        return name;
    }
    public Address getAddress()
    {
        return address;
    }
    public String getPhoneNumber()
    {
        return phoneNumber;
    }
    public String getEmail()
    {
        return email;
    }
    public void setName(String name)
    {
        this.name = name;
    }
    public void setAddress(Address address)
    {
        this.address = address;
    }
    public void setPhoneNumber(String phoneNumber)
    {
        this.phoneNumber = phoneNumber;
    }
    public void setEmail(String email)
    {
        this.email = email;
    }
    @Override
    public String toString()
    {
        return this.getClass() + "\nname: " + name;
    }
}
```

- Employee

```
import java.time.LocalDate;
public class Employee extends Person
{
    protected String office;
    protected double salary;
    protected LocalDate dateHired;
    public Employee() {}
    public Employee(String n, Address a, String p, String e, String o, double s)
    {
        super(n, a, p, e);
        office = o;
        salary = s;
        dateHired = LocalDate.now();
    }
    public String getOffice()
    {
        return office;
    }
    public double getSalary()
    {
        return salary;
    }
    public LocalDate getDateHired()
    {
        return dateHired;
    }
    public void setOffice(String office)
    {
        this.office = office;
    }
    public void setSalary(double salary)
    {
        this.salary = salary;
    }
    @Override
    public String toString()
    {
        return this.getClass() + "\nname: " + name; //super.toString();
    }
}
```

- Student

```
public class Student extends Person
{
    private final static String STATUS1 = "freshman";
    private final static String STATUS2 = "sophomore";
    private final static String STATUS3 = "junior";
    private final static String STATUS4 = "senior";
    private String status;
    public Student(){}
    public Student(String n, Address a, String p, String e, String s)
    {
        super(n, a, p, e);
        status = s;
    }
    public static String getSTATUS1()
    {
        return STATUS1;
    }
    public static String getSTATUS2()
    {
        return STATUS2;
    }
    public static String getSTATUS3()
    {
        return STATUS3;
    }
    public static String getSTATUS4()
    {
        return STATUS4;
    }
    public void setStatus(String status)
    {
        this.status = status;
    }
    @Override
    public String toString()
    {
        return this.getClass() + "\nname: " + name;
    }
}
```

- Staff

```
import java.time.LocalDate;
public class Employee extends Person
{
    protected String office;
    protected double salary;
    protected LocalDate dateHired;
    public Employee(String n, Address a, String p, String e, String o, double s)
    {
        super(n, a, p, e);
        office = o;
        salary = s;
        dateHired = LocalDate.now();
    }
    public String getOffice()
    {
        return office;
    }
    public double getSalary()
    {
        return salary;
    }
    public LocalDate getDateHired()
    {
        return dateHired;
    }
    public void setOffice(String office)
    {
        this.office = office;
    }
    public void setSalary(double salary)
    {
        this.salary = salary;
    }
    @Override
    public String toString()
    {
        return this.getClass() + "\nname: " + name;
    }
}
```

- Faculty

```
public class Faculty extends Employee
{
    private String officeHours;
    private String rank;
    public Faculty() {}
    public Faculty(String n, Address a, String p, String e, String o, double s, String oh, String
        r)
    {
        super(n, a, p, e, o, s);
        officeHours = oh;
        rank = r;
    }
    public String getOfficeHours()
    {
        return officeHours;
    }
    public String getRank()
    {
        return rank;
    }
    public void setOfficeHours(String officeHours)
    {
        this.officeHours = officeHours;
    }
    public void setRank(String rank)
    {
        this.rank = rank;
    }
    @Override
    public String toString()
    {
        return this.getClass() + "\nname: " + name;
    }
}
```

- Address

```
public class Address
{
    private String country;
    private String city;
    private String street;
    public Address(String country, String city, String street)
    {
        this.country = country;
        this.city = city;
        this.street = street;
    }
    public String getCountry()
    {
        return country;
    }
    public String getCity()
    {
        return city;
    }
    public String getStreet()
    {
        return city;
    }
    public void setCountry(String country)
    {
        this.country = country;
    }
    public void setCity(String city)
    {
        this.city = city;
    }
    public void setStreet(String street)
    {
        this.street = street;
    }
    public String toString()
    {
        return "\nCountry: " + country + "\nCity: " + city + "\nStreet: " + street;
    }
}
```

- Test

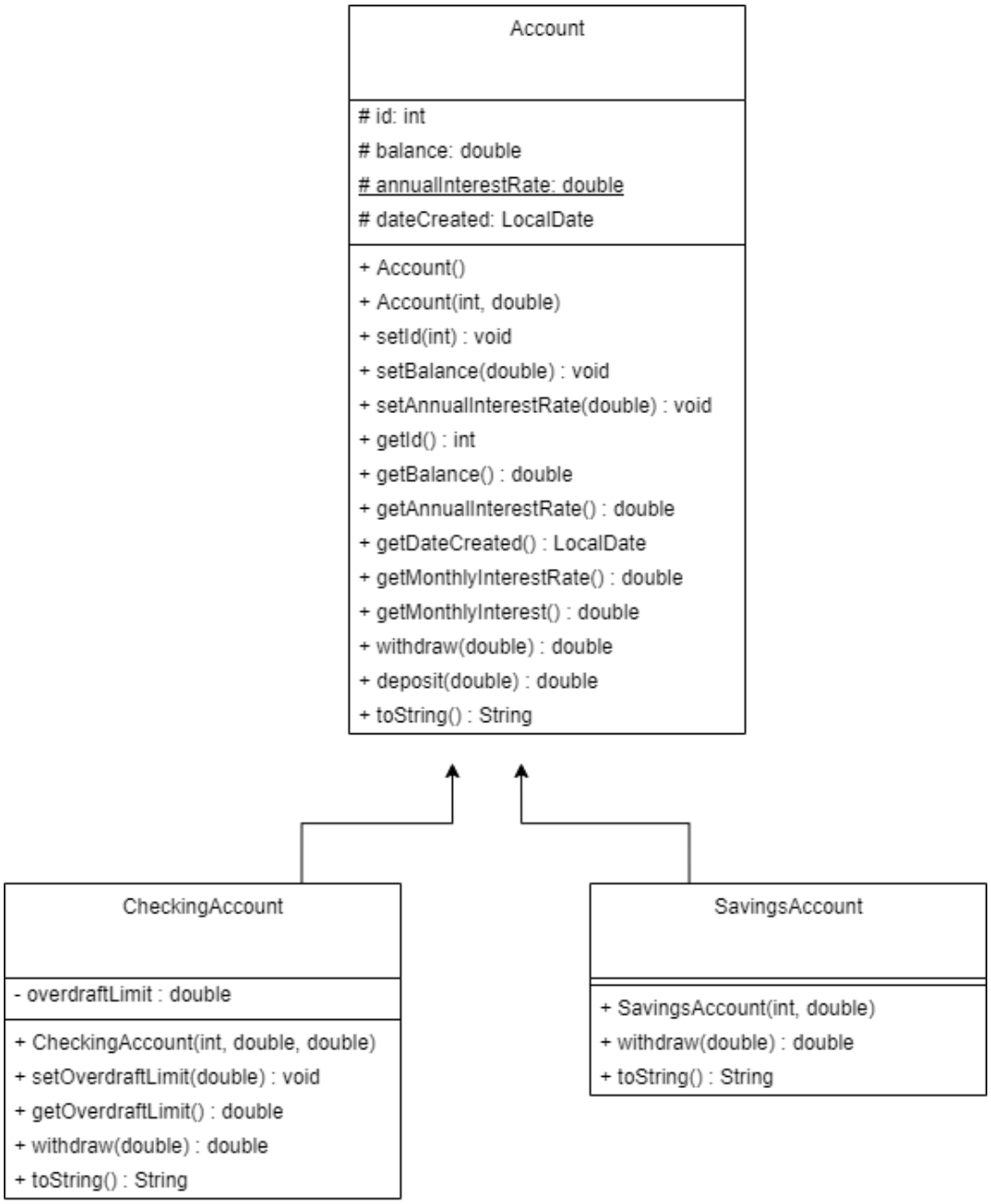
```
public class Test
{
    public static void main(String[] args)
    {
        Person p = new Person("Reem", new Address("SA", "Jeddah", "Street"), "0506321951",
            "Reem@gmail.com");
        System.out.println(p.toString());

        Student s = new Student("Leen", new Address("SA", "Jeddah", "Faisalia"), "0512345678",
            "Leen@effat.edu.sa", Student.getSTATUS1());
        System.out.println("\n" + s.toString());

        Employee e = new Employee("Akila", new Address("SA", "Jeddah", "Khuzam"), "0501223461",
            "Akila@effatuniversity.edu.sa", "CoE145", 10000000);
        System.out.println("\n" + e.toString());

        Faculty f = new Faculty("Moh", new Address("SA", "Jeddah", "Obhur"), "0598776310",
            "Moh@effatuniversity.edu.sa", "A&R313", 100000, "12:30 - 2:20", "prof");
        System.out.println("\n" + f.toString());

        Staff staff = new Staff("Dana", new Address("SA", "Jeddah", "Rawda"), "0587927105",
            "Dana@effatuniversity.edu.sa", "A&R304", 5000, "instructor");
        System.out.println("\n" + staff.toString());

    }
}
```

- Output

```
<terminated> Test (5) [Java Application] C:\Users\reemH\C
class Person
name: Reem

class Student
name: Leen

class Employee
name: Akila

class Faculty
name: Moh

class Staff
name: Dana
```

2.2 Question 2

- UML



- Account

```
import java.time.LocalDate;
public class Account
{
    protected int id;
    protected double balance;
    protected static double annualInterestRate;
    protected LocalDate dateCreated;
    public Account()
    {
        id = 0;
        balance = 0;
        annualInterestRate = 0;
        dateCreated = LocalDate.now();
    }
    public Account(int id, double balance)
    {
        this.id = id;
        this.balance = balance;
        dateCreated = LocalDate.now();
    }
    public void setId(int id)
    {
        this.id = id;
    }
    public void setBalance(double balance)
    {
        this.balance = balance;
    }
    public void setAnnualInterestRate(double annualInterestRate)
    {
        this.annualInterestRate = annualInterestRate;
    }
    public int getId()
    {
        return id;
    }
    public double getBalance()
    {
        return balance;
    }
    public double getAnnualInterestRate()
    {
        return annualInterestRate;
    }
    public LocalDate getDateCreated()
    {
        return dateCreated;
    }
    public double getMonthlyInterestRate()
    {
        return (annualInterestRate/12)/100;
    }
    public double getMonthlyInterest()
    {
        return balance*getMonthlyInterestRate();
    }
    public double withdraw(double amount)
```

```
    {
        return balance-=amount;
    }
    public double deposit(double amount)
    {
        return balance+=amount;
    }
    @Override
    public String toString()
    {
        return this.getClass() + "\nID: " + id + "\nBalance: " + balance + "\nAnnual Interest Rate:
            " + annualInterestRate + "\nDate Created: " + dateCreated
               + "\nMonthly Interest Rate: " + getMonthlyInterestRate() + "%\nMonthly Interest" +
                   getMonthlyInterest();
    }
}
```

- CheckingAccount

```
public class CheckingAccount extends Account
{
    private double overdraftLimit;
    public CheckingAccount()
    {
        super();
    }
    public CheckingAccount(int id, double balance, double overdraftLimit)
    {
        super(id, balance);
        this.overdraftLimit = overdraftLimit;
    }
    public void setOverdraftLimit(double overdraftLimit)
    {
        this.overdraftLimit = overdraftLimit;
    }
    public double getOverdraftLimit()
    {
        return overdraftLimit;
    }
    @Override
    public double withdraw(double amount)
    {
        if (balance - amount < overdraftLimit)
            return balance;
        else
            return balance-=amount;
    }
    @Override
    public String toString()
    {
        return super.toString() + "\nOverdraft Limit: " + overdraftLimit;
    }
}
```

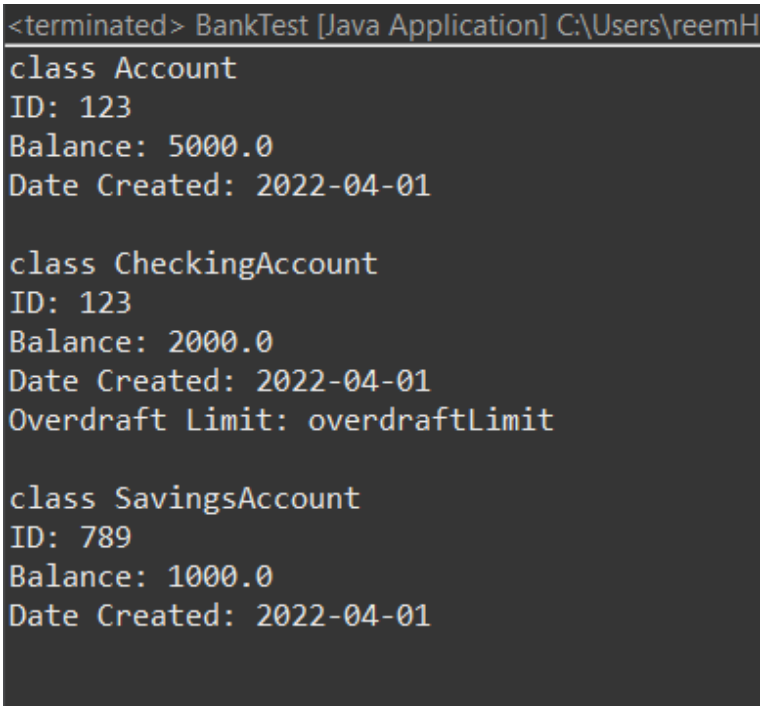
- SavingsAccount

```
public class SavingsAccount extends Account
{
    public SavingsAccount()
    {
        super();
    }
    public SavingsAccount(int id, double balance)
    {
        super(id, balance);
    }
    @Override
    public double withdraw(double amount)
    {
        if (balance < amount)
            return balance;
        else
            return balance-=amount;
    }
    @Override
    public String toString()
    {
        return this.getClass() + "\nID: " + id + "\nBalance: " + balance + "\nDate Created: " +
            dateCreated;
    }
}
```

- Test

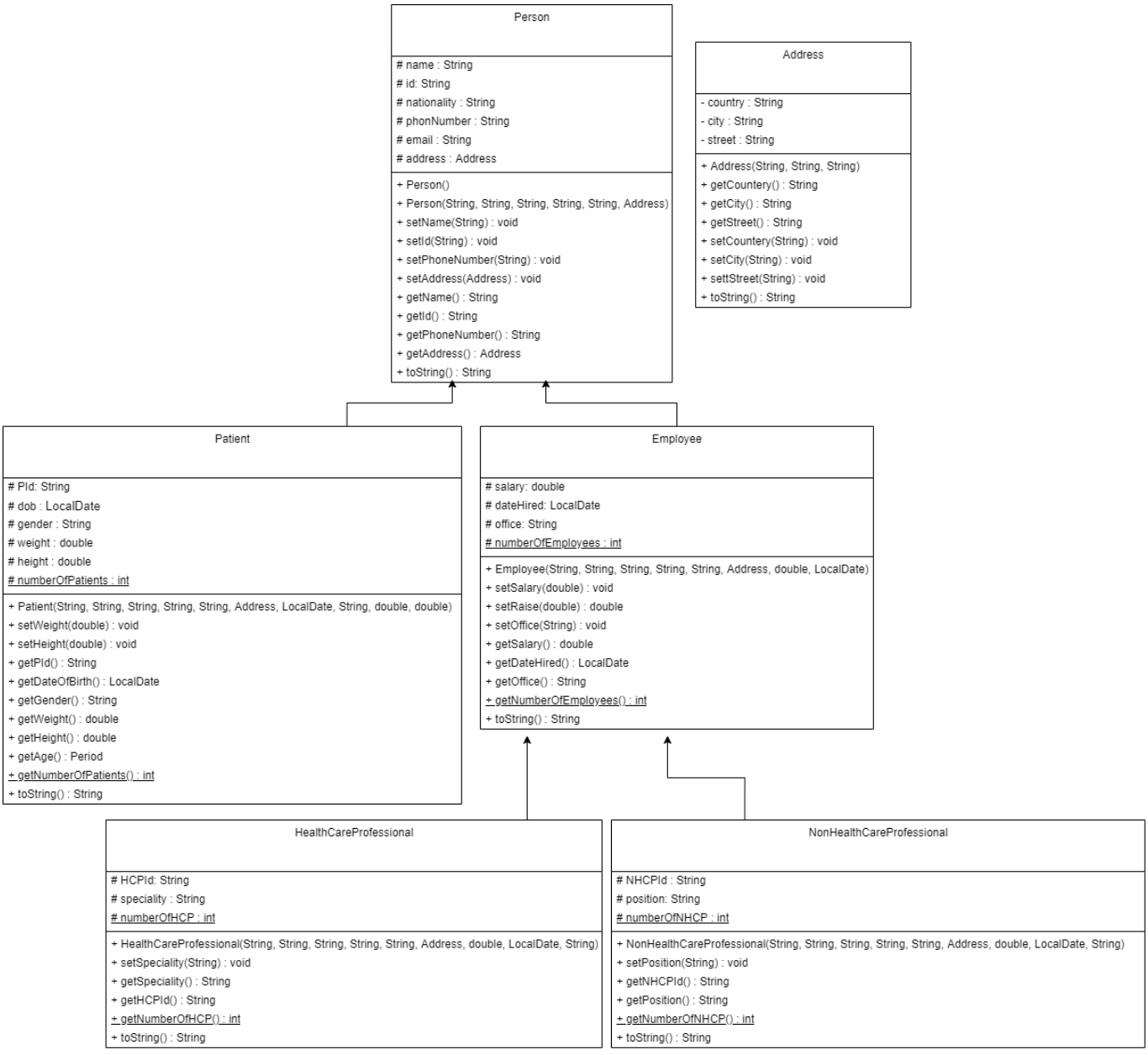
```
public class BankTest
{
    public static void main(String[] args)
    {
        Account a = new Account(123, 5000);
        System.out.println(a.toString() + "\n");

        CheckingAccount c = new CheckingAccount(123, 2000, 1000);
        System.out.println(c.toString() + "\n");

        SavingsAccount s = new SavingsAccount(789, 1000);
        System.out.println(s.toString());
    }
}
```

- Output



2.3 Question 3

- UML



- Person

```
public class Person
{
    protected String name;
    protected String nId;
    protected String nationality;
    protected String phoneNumber;
    protected String email;
    protected Address address;
    public Person() {}
    public Person(String name, String nId, String nationality, String phoneNumber, String email,
         Address address)
    {
        this.name = name;
        this.nId = nId;
        this.nationality = nationality;
        this.phoneNumber = phoneNumber;
        this.email = email;
        this.address = address;
    }
    public void setName(String name)
    {
        this.name = name;
    }
    public void setNId(String nId)
    {
        this.nId = nId;
    }
    public void setNationality(String nationality)
    {
        this.nationality = nationality;
    }
    public void setPhoneNumber(String phoneNumber)
    {
        this.phoneNumber = phoneNumber;
    }
    public void setEmail(String email)
    {
        this.email = email;
    }
    public void setAddress(Address address)
    {
        this.address = address;
    }
    public String getName()
    {
        return name;
    }
    public Address getAddress()
    {
        return address;
    }
    public String getPhoneNumber()
    {
        return phoneNumber;
    }
    public String getEmail()
    {
```

```
        return email;
    }
    @Override
    public String toString()
    {
        return this.getClass() + "\nname: " + name;
    }
}
```

- Patient

```
import java.time.LocalDate;
import java.time.Period;
import java.util.UUID;
public class Patient extends Person
{
    protected String PId;
    protected LocalDate dob;
    protected String gender;
    protected double weight;
    protected double height;
    protected static int numberOfPatients;

    Patient(String name, String nId, String nationality, String phoneNumber, String email, Address
        address,
           LocalDate dob, String gender, double weight, double height)
    {
        super(name, nId, nationality, phoneNumber, email, address);
        PId = UUID.randomUUID().toString();
        this.dob = dob;
        this.gender = gender;
        this.weight = weight;
        this.height = height;
        numberOfPatients += 1;
    }
    public void setWeight(double weight)
    {
        this.weight = weight;
    }
    public void setHeight(double height)
    {
        this.height = height;
    }
    public String getPId()
    {
        return PId;
    }
    public LocalDate getDateOfBirth()
    {
        return dob;
    }
    public String getGender()
    {
        return gender;
    }
    public double getWeight()
    {
        return weight;
    }
    public double getHeight()
    {
        return height;
    }
    public Period getAge()
    {
        LocalDate today = LocalDate.now();
        return Period.between(dob, today);
    }
```

```
    public static int getNumberOfPatients()
    {
        return numberOfPatients;
    }
    @Override
    public String toString()
    {
        return this.getClass() + "\nname: " + name;
    }
}
```

- Employee

```
import java.time.LocalDate;
public class Employee extends Person
{
    protected double salary;
    protected String office;
    protected LocalDate dateHired;
    protected static int numberOfEmployees;

    public Employee(String name, String nId, String nationality, String phoneNumber, String email,
         Address address,
            double salary, String office)
    {
        super(name, nId, nationality, phoneNumber, email, address);
        this.salary = salary;
        this.office = office;
        dateHired = LocalDate.now();
        numberOfEmployees += 1;
    }
    public void setSalary(double salary)
    {
        this.salary = salary;
    }
    public void setRaise(double raise)
    {
        salary += raise;
    }
    public void setOffice(String office)
    {
        this.office = office;
    }
    public double getSalary()
    {
        return salary;
    }
    public LocalDate getDateHired()
    {
        return dateHired;
    }
    public String getOffice()
    {
        return office;
    }
    public static int getNumberOfEmployees()
    {
        return numberOfEmployees;
    }
    @Override
    public String toString()
    {
        return this.getClass() + "\nname: " + name;
    }
}
```

- HealthCareProfessional

```
import java.util.UUID;
public class HealthCareProfessional extends Employee
{
    protected String HCPId;
    protected String speciality;
    protected static int numberOfHCP;

    public HealthCareProfessional(String name, String nId, String nationality, String phoneNumber,
         String email,
            Address address, double salary, String office, String HCPId, String speciality)
    {
        super(name, nId, nationality, phoneNumber, email, address, salary, office);
        HCPId = UUID.randomUUID().toString();
        this.speciality = speciality;
        numberOfHCP += 1;
    }
    public void setSpeciality(String speciality)
    {
        this.speciality = speciality;
    }
    public String getHCPId()
    {
        return HCPId;
    }
    public String getSpeciality()
    {
        return speciality;
    }
    public static int getNumberOfHCP()
    {
        return numberOfHCP;
    }
    @Override
    public String toString()
    {
        return this.getClass() + "\nname: " + name;
    }
}
```

- NonHealthCareProfessional

```
import java.util.UUID;

public class NonHealthCareProfessional extends Employee
{
    protected String NHCPId;
    protected String position;
    protected static int numberOfNHCP;

    public NonHealthCareProfessional(String name, String nId, String nationality, String
        phoneNumber, String email,
           Address address, double salary, String office, String NHCPId, String position)
    {
        super(name, nId, nationality, phoneNumber, email, address, salary, office);
        NHCPId = UUID.randomUUID().toString();
        this.position = position;
        numberOfNHCP += 1;
    }
    public void setPosition(String position)
    {
        this.position = position;
    }
    public String getNHCPId()
    {
        return NHCPId;
    }
    public String getSpeciality()
    {
        return position;
    }
    public static int getNumberOfNHCP()
    {
        return numberOfNHCP;
    }
    @Override
    public String toString()
    {
        return this.getClass() + "\nname: " + name;
    }
}
```

- Test

```
import java.time.LocalDate;
import java.time.Month;

public class Test
{
    public static void main(String []args)
    {
        Person person = new Person("Leen", "1123", "Saudi", "0512345678", "Leen@gmail.com", new
            Address("SA", "Jeddah", "Alshatee"));
        Patient patient = new Patient("Nada", "12345", "Saudi", "0598765310", "Nada@hotmail.com",
            new Address("SA", "Jeddah", "AlRawda"), LocalDate.of(1980, Month.JANUARY, 1),
            "Female", 60, 170);
        Employee employee = new Employee("Ahmad", "12344", "Saudi", "0518239583",
            "Ahmad@gmail.com", new Address("SA", "Jeddah", "AlFaisalia"), 30000, "B1-108");
        HealthCareProfessional d = new HealthCareProfessional("Reem", "1223", "Syrian",
            "0550632177", "Reem@gmail.com", new Address("SA", "Jeddah", "AlNuzha"), 999999999,
            "B1-145", "NET");
        NonHealthCareProfessional r = new NonHealthCareProfessional("Tala", "1958", "Saudi",
            "0506793842", "Tala@gmail.com", new Address("SA", "Jeddah", "Obhur"), 10000, "Lobby",
            "Receptionist");
        System.out.print(person.toString() + "\n" + "\n" + patient.toString() + "\n" + "\n" +
            employee.toString() + "\n"
        + "\n" + d.toString() + "\n" + "\n" + r.toString());
    }
}
```

- Output

```
class Person
name: Leen

class Patient
name: Nada

class Employee
name: Ahmad

class HealthCareProfessional
name: Reem

class NonHealthCareProfessional
name: Tala
```

3 Conclusion

This lab was very long, but the questions are easy, and it was good to start working on the project.