20 june 2019

Account class

```
public class Account {
	int accountid;
	int customerid;
	String accountType;
	double AccountBalance;

	public Account(int accountid, int customerid, String accountType, double
accountBalance) {
		super();
		this.accountid = accountid;
		this.customerid = customerid;
		this.accountType = accountType;
		AccountBalance = accountBalance;
	}
	public Account() {}
	public int getAccountid() {
		return accountid;
	}
	public void setAccountid(int accountid) {
		this.accountid = accountid;
	}
	public int getCustomerid() {
		return customerid;
	}
	public void setCustomerid(int customerid) {
		this.customerid = customerid;
	}
	public String getAccountType() {
		return accountType;
	}
	public void setAccountType(String accountType) {
		this.accountType = accountType;
	}
	public double getAccountBalance() {
		return AccountBalance;
	}
	public void setAccountBalance(double accountBalance) {
		AccountBalance = accountBalance;
	}

}
```

Class Account demo

```
package Accountt;

public class Accountdemo {

	public static double deductInterest(Account acc,double deduct) {
		double result= acc.getAccountBalance()-
((acc.getAccountBalance()/100)*deduct);
		acc.setAccountBalance(result);
		return result;
	}

	public static Account[] getAccountwithmorethanAvgBalance(Account[] acc1) {
		double avg=0;
		for(Account ac:acc1) {
			avg = ac.getAccountBalance()+avg;
		}
		avg=avg/acc1.length;
		int count=0,i=0;
		for(Account ac:acc1) {
			if(ac.getAccountBalance()>avg) {
				count++;
			}
		}
		Account[] result = new Account[count];
		for(Account ac:acc1) {
			if(ac.getAccountBalance()>avg) {
				result[i]=ac;
				i++;
			}
		}
		return result;
		}


	public static void main(String[] args) {
		Account[] accounts = new Account[5];
		accounts[0]= new Account(1,1,"SA",10000);
		accounts[1]= new Account(2,2,"CA",20000);
		accounts[2]= new Account(3,1,"SA",30000);
		accounts[3]= new Account(4,2,"CA",40000);
		accounts[4]= new Account(5,3,"SA",50000);
		System.out.println(deductInterest(accounts[0],10.0));
		System.out.println(accounts[0].getAccountBalance());
		Account[] acc = new Account[3];
		acc=getAccountwithmorethanAvgBalance(accounts);
		for(int i=0;i<acc.length;i++) {
			System.out.println(acc[i].getAccountid());
		}
	}
}
```

PRA Appartment set1

Class Appartment

```
public class Apartment {
	int flatNo;
	String flatOwner;
	double Water_bill;
	double Electricity_bill;

	public Apartment(int flatNo, String flatOwner, double water_bill, double
electricity_bill) {
		super();
		this.flatNo = flatNo;
		this.flatOwner = flatOwner;
		Water_bill = water_bill;
		Electricity_bill = electricity_bill;
	}

	public int getFlatNo() {
		return flatNo;
	}
	public void setFlatNo(int flatNo) {
		this.flatNo = flatNo;
	}
	public String getFlatOwner() {
		return flatOwner;
	}
	public void setFlatOwner(String flatOwner) {
		this.flatOwner = flatOwner;
	}
	public double getWater_bill() {
		return Water_bill;
	}
	public void setWater_bill(double water_bill) {
		Water_bill = water_bill;
	}
	public double getElectricity_bill() {
		return Electricity_bill;
	}
	public void setElectricity_bill(double electricity_bill) {
		Electricity_bill = electricity_bill;
	}
}
```

Class Appartment demo

```
package Apartmentt;

public class Apartmentdemo {

	public static double getsecondwaterbill(Apartment[] obj) {
		for(int i=0;i<obj.length;i++) {
			for(int j=i+1;j<obj.length;j++) {
				if(obj[i].getWater_bill()>obj[j].getWater_bill()) {
					Apartment temp = obj[i];
					obj[i]=obj[j];
					obj[j]=temp;
				}
			}
		}
		return obj[1].getWater_bill();
	}

	public static String getflatOwnerofminelectricitybill(Apartment[] obj) {
		for(int i=0;i<obj.length;i++) {
			for(int j=i+1;j<obj.length-1-i;j++) {

	if(obj[i].getElectricity_bill()>obj[j].getElectricity_bill()) {
					Apartment temp = obj[i];
					obj[i]=obj[j];
					obj[j]=temp;
				}
			}
		}
		return obj[0].getFlatOwner();
	}

	public static void main(String[] args) {
		Apartment a1 = new Apartment(101,"Ram",400,660);
		Apartment a2 = new Apartment(102,"Krish",420,650);
		Apartment a3 = new Apartment(103,"Vijay",430,630);
		Apartment a4 = new Apartment(104,"Murthy",480,690);
		Apartment a5 = new Apartment(105,"Jhon",460,680);
		Apartment[] apt= {a1,a2,a3,a4,a5};
		System.out.println("Second min water bill :"+ getsecondwaterbill(apt));
		System.out.println("Flat Owner of Min Electricity bill :" +
getflatOwnerofminelectricitybill(apt));
	}

}
```

PRA iASCERT

Class Employee

```
public class Employee {
	private int empid;
	private String empName;
	private double salary;
	private String grade;
	private String location;

	public Employee(int empid, String empName, double salary, String grade, String
location) {
		super();
		this.empid = empid;
		this.empName = empName;
		this.salary = salary;
		this.grade = grade;
		this.location = location;
	}
	public int getEmpid() {
		return empid;
	}
	public void setEmpid(int empid) {
		this.empid = empid;
	}
	public String getEmpName() {
		return empName;
	}
	public void setEmpName(String empName) {
		this.empName = empName;
	}
	public double getSalary() {
		return salary;
	}
	public void setSalary(double salary) {
		this.salary = salary;
	}
	public String getGrade() {
		return grade;
	}
	public void setGrade(String grade) {
		this.grade = grade;
	}
	public String getLocation() {
		return location;
	}
	public void setLocation(String location) {
		this.location = location;
	}


}
```

Class employeedemo

```
public class Employeedemo {
	private static Employee[] incrementSalaryofEmployees(Employee[] emp){
		for(Employee e:emp) {
		if(e.getGrade().equals("ASE")) {
			e.setSalary(e.getSalary()+((e.getSalary()/10) * 3));
			}
		if(e.getGrade()=="SE") {
			e.setSalary(e.getSalary()+((e.getSalary()/10) * 5));
			}
		if(e.getGrade()=="ITA") {
			e.setSalary(e.getSalary()+((e.getSalary()/10) * 7));
			}
		if(e.getGrade()=="AST") {
			e.setSalary(e.getSalary()+((e.getSalary()/10) * 9));
			}
		}
		return emp;
		}
	public static int numberofemployees(Employee[] emp,String loc) {
		int count=0;
		for(Employee e:emp) {
			if(e.getLocation().equals(loc)) {
				count++;
			}
		}
		return count;
	}
	public static void main(String[] args) {
		Employee e1 = new Employee(1,"abc",20000.00,"AST","MUM");
		Employee e2 = new Employee(2,"def",25000.00,"ITA","DEL");
		Employee e3 = new Employee(3,"ghi",30000.00,"ITA","MUM");
		Employee e4 = new Employee(4,"jkl",27000.00,"ASE","CHN");
		Employee e5 = new Employee(5,"mno",53000.00,"SE","HYD");

		Employee[] emp = new Employee[] {e1,e2,e3,e4,e5};
		int empcount=numberofemployees(emp,"MUM");
		System.out.println("Total no of employee in MUM is : " + empcount);

		Employee updatedsalary[] = Employeedemo.incrementSalaryofEmployees(emp);

		for(Employee e:updatedsalary) {
		System.out.println(e.getEmpid()+" "+e.getEmpName()+" "+e.getGrade()+"
"+e.getLocation()+" "+e.getSalary());
		}
	}
}
```

PRA_question

Class food

```java
public class Food {
	int foodId;
	String Name;
	double price;
	int quantity;


	public Food(int foodId, String name, double price, int quantity) {
		super();
		this.foodId = foodId;
		Name = name;
		this.price = price;
		this.quantity = quantity;
	}
	public int getFoodId() {
		return foodId;
	}
	public void setFoodId(int foodId) {
		this.foodId = foodId;
	}
	public String getName() {
		return Name;
	}
	public void setName(String name) {
		Name = name;
	}
	public double getPrice() {
		return price;
	}
	public void setPrice(double price) {
		this.price = price;
	}
	public int getQuantity() {
		return quantity;
	}
	public void setQuality(int quantity) {
		this.quantity = quantity;
	}
}
```

```
public class Restaurentdemo {
	public static Food[] findfoodbyQuantity(int y,Food[] food) {
		int count=0;
		for(Food f:food) {
			if(f.getQuantity()>y) {
				count++;
			}
		}
		int i=0;
		Food[] result = new Food[count];
		for(Food f:food) {
		if(f.getQuantity()>y) {
			result[i] = f;
			i++;
			}
		}
		return result;
	}

	public static Food findfoodwithHighestbill(Food[] food) {
		for(int i=0;i<food.length;i++) {
			for(int j=i+1;j<food.length-i-1;j++) {

	if((food[i].getPrice()*food[i].getQuantity())<(food[j].getPrice()*food[j].getQ
uantity())) {
					Food temp = food[i];
					food[i]=food[j];
					food[j]=temp;
				}
			}
		}
		return food[0];
	}
	public static void main(String[] args) {
		Food r1= new Food(12,"Biryani",80,5);
		Food r2= new Food(13,"Ice Cream",50,9);
		Food r3= new Food(14,"Gulab Jamun",10,15);
		Food r4= new Food(15,"Chicken Labavdar",150,7);
		Food r5= new Food(16,"Butter Naan",20,8);

		Food res[] = {r1,r2,r3,r4,r5};
		Food res1[] = findfoodbyQuantity(7,res);
		for(int i=0;i<res1.length;i++) {
			System.out.println(res1[i].getFoodId()+" "+res1[i].getName()+"
"+res1[i].getPrice()+" "+res1[i].getQuantity());
		}
		Food r=findfoodwithHighestbill(res);
		System.out.println(r.getFoodId()+" "+r.getName()+" "+r.getPrice()+"
"+r.getQuantity());
	}

}
```