OOP FINAL EXAM PAPER ANSWERS 2018

Question 01.

Book.java

```
package Ques01;

public class Book extends Item {

	private String publisher;
	private String category;
	private int pages;

	public Book(int itemNumber, String description, double unitPrice, String
publisher, String category, int pages) {
		super(itemNumber, description, unitPrice);
		this.category = category;
		this.pages = pages;
		this.publisher = publisher;
	}

	@Override
	public void display() {
		super.display();
		System.out.println("Publisher : " + publisher + "\nCategory : " +
category + "\nPages : " + pages + "\n");
	}



}
```

Car.java

```
	package Ques01;

public class Car extends Item{

	private String model;
	private String type;

	public Car(int itemNumber, String description, double unitPrice, String
model, String type) {
		super(itemNumber, description, unitPrice);
		this.model = model;
		this.type = type;
	}

	@Override
```

```
	public void display() {
		super.display();
		System.out.println("Model : " + model + "\nType : " + type + "\n");
	}



}
```

Item.java

```
package Ques01;

public abstract class Item {
	private int itemNumber;
	private String description;
	private double unitPrice;

	public Item(int itemNumber, String description, double unitPrice) {
		super();
		this.itemNumber = itemNumber;
		this.description = description;
		this.unitPrice = unitPrice;
	}

	public void display() {
		System.out.println("Item number : " + itemNumber + "\nDescription :
" + description + "\nUnit Price : " + unitPrice);
	}


}
```

MainApp.java

```
package Ques01;

import java.util.ArrayList;

public class MainApp {

	public static void main(String[] args) {
		ArrayList<Item> items = new ArrayList<>();
		Item b1 = new Book(1, "D1", 500.00, "P1", "C1", 200);
		Item b2 = new Book(2, "D2", 600.00, "P2", "C2", 200);

		items.add(b1);
		items.add(b2);

		Item c1 = new Car(3, "D3", 2500000, "m1", "t1");
		Item c2 = new Car(4, "D4", 4900000, "m4", "t4");

		items.add(c1);
```

```
		items.add(c2);

		for(Item i : items) {
			i.display();
		}
	}

}
```

Question2a

CalcSum.java

```
package Ques02;

public class CalcSum implements Runnable{
	private long sum = 1;
	@Override
	public void run() {

		synchronized(CalcSum.class) {
			for(int i = 0; i < 100000; i++) {
				sum += i;
			}

			System.out.println("Current Thread : "+
Thread.currentThread().getName() + "Sum : " + sum);
		}
	}

}
```

CountDown.java

```
package Ques02;

public class CountDown extends Thread{
	public void run() {
		synchronized(CountDown.class) {
			for(int i = 0; i < 10; i++) {
				System.out.println(i+1);
				try {
					sleep(1000);
				} catch (InterruptedException e) {
					// TODO Auto-generated catch block
```

```
                                e.printStackTrace();
                            }
                        }
                    }
                }
}
```

MainThreadApp.java

```
package Ques02;

public class MainThreadApp {

	public static void main(String[] args) throws InterruptedException  {
		// TODO Auto-generated method stub
		Thread countDown = new Thread(new CountDown());

		Thread calcSum1 = new Thread(new CalcSum());
		Thread calcSum2 = new Thread(new CalcSum());

		calcSum1.setName("Black");
		calcSum2.setName("White");

		countDown.start();
		countDown.join();


		calcSum1.start();
		calcSum1.join();

		calcSum2.start();
		calcSum2.join();

	}

}
```

Question2b

Calculation.java

```
package Ques2b;

public class Calculation {
	private double ans = 1;
```

```
	public double getAns() {
		return ans;
	}

	void Factorial(int start, int end) {

		System.out.println("Compution start : " + start + " to end : " +
end);
		for(int i = start ; i <= end; i++) {
			ans *= (double)i;
		}

	}
}
```

ParallelThread.java

```
package Ques2b;

public class ParallelThread implements Runnable{
	private Calculation myCalc;
	private int start;
	private int end;

	public ParallelThread(Calculation myCalc, int start, int end) {
		super();
		this.myCalc = myCalc;
		this.start = start;
		this.end = end;
	}

	@Override
	public void run() {
		// TODO Auto-generated method stub
		synchronized(ParallelThread.class) {
			myCalc.Factorial(start, end);
		}
	}

}
```

MainThreadApp.java

```
package Ques2b;

public class MainThreadApp {
	public static void main(String[] args) throws InterruptedException {
```

```
		Calculation calc = new Calculation();

		Thread t1 = new Thread(new ParallelThread(calc, 1, 20));
		Thread t2 = new Thread(new ParallelThread(calc, 21, 40));

		t1.start();
		t1.join();

		t2.start();
		t2.join();

		System.out.println("Answer is : " + calc.getAns());

	}
}
```

Question03

Students.java

```
package Ques03;

import java.util.ArrayList;
import java.util.Scanner;

public class Students {
	private int id;
	private String name;
	private ArrayList<Float> marks = new ArrayList<>();
	private int noOfSubjects;

	public Students(int id, String name) {
		super();
		this.id = id;
		this.name = name;
	}

	public float inputMarks(int index) throws MarksException{
		Scanner sc = new Scanner(System.in);
```

```
            System.out.print("Enter Mark : ");
            float i = sc.nextFloat();
            marks.add(i);

            if(marks.get(index) >= 0 && marks.get(index) <= 100) {
                  return marks.get(index);
            }else {
                  marks.remove(index);
                  throw new MarksException(marks.get(index));
            }


      }

      public void input() {
            Scanner sc = new Scanner(System.in);

            System.out.print("Enter number of subjects : ");
            noOfSubjects = sc.nextInt();

            try {

                  for(int i = 0; i < noOfSubjects; i++) {
                        inputMarks(i);
                  }

            } catch (MarksException e) {
                  // TODO Auto-generated catch block
                  e.printStackTrace();
            }

      }

      public float getAverage() {
```

```
            float total = 0.0f;
            int count = 0;
            float average = 0;

            for(float t : marks) {
                  total += t;
                  count++;
            }

            try {
                  average = total / count;
            }catch(ArithmeticException e) {
                  e.printStackTrace();
            }

            return average;


      }
}
```

MarksException.java

```
package Ques03;

public class MarksException extends Throwable{

      private float marks;

      public MarksException(float marks) {
            super();
            this.marks = marks;
      }

      public float getMarks() {
            return marks;
      }

}
```

MainApp.java

```
package Ques03;

public class MainApp {

	public static void main(String[] args) {
		Students s = new Students(1, "Yugma");

		s.input();
		System.out.println(s.getAverage());
	}

}
```

Question4a

CreditCard.java

```
package Ques4a;

public class CreditCard {
	public static CreditCard c = null;

	private CreditCard() {}

	public static CreditCard getObject() {

		if(c == null) {
			c = new CreditCard();
			System.out.println("Creating new object!");
		}else {
			System.out.println("Returning existing object!");
		}

		return c;
	}

	public boolean validate(String cardNo, int code) {
		int remainder = code % 3;

		if(cardNo.length() == 16 && remainder == 0) {
			return true;
		}else {
			return false;
		}
	}
}
```

MianApp.java

```
package Ques4a;

public class MianApp {
	public static void main(String[] args) {
		CreditCard c1 = CreditCard.getObject();
		CreditCard c2 = CreditCard.getObject();

		System.out.println(c1.validate("IT17167192178250", 3));
		System.out.println(c2.validate("IT17167192178250", 9));

	}
}
```

Question4b

Command.java

```
package Ques4b;

public interface Command {
	public void execute();
}
```

GarageGate.java

```
package Ques4b;

public class GarageGate {
	private String description;

	public GarageGate(String description) {
		super();
		this.description = description;
	}

	public void Open() {
		System.out.println(description + " Opening!");
	}

	public void Close() {
		System.out.println(description + " Closing!");
```

```
	}
}
```

GarageGateClose.java

```
package Ques4b;

public class GarageGateClose implements Command{

	private GarageGate g;


	public GarageGateClose(GarageGate g) {
		super();
		this.g = g;
	}


	@Override
	public void execute() {
		// TODO Auto-generated method stub
		g.Close();
	}

}
```

GarageGateOpen.java

```
package Ques4b;

public class GarageGateOpen implements Command{
	private GarageGate g;


	public GarageGateOpen(GarageGate g) {
		super();
		this.g = g;
	}


	@Override
	public void execute() {
		// TODO Auto-generated method stub
		g.Open();
```

```
	}

}
```

Oven.java

```
package Ques4b;

public class Oven {
	private String name;

	public Oven(String name) {
		super();
		this.name = name;
	}

	public void On() {
		System.out.println(name + " Switching on!");
	}

	public void Off() {
		System.out.println(name + " Switching off!");
	}
}
```

OvenOn.java

```
package Ques4b;

public class OvenOn implements Command{
	private Oven o;


	public OvenOn(Oven o) {
		super();
		this.o = o;
	}


	@Override
	public void execute() {
		// TODO Auto-generated method stub
		o.On();
	}

}
```

OvenOff.java

```java
package Ques4b;

public class OvenOff implements Command{

	private Oven o;



	public OvenOff(Oven o) {
		super();
		this.o = o;
	}



	@Override
	public void execute() {
		// TODO Auto-generated method stub
		o.Off();
	}
}
```

MobileUI.java

```java
package Ques4b;

public class MobileUI {

	private Command commands[];

	public MobileUI() {
		super();
		commands = new Command[6];
	}

	public void setCommand(int index, Command cmdObj) {
		commands[index] = cmdObj;
	}

	public void commmandPressed(int index) {
		commands[index].execute();
	}
}
```

MainApp.java

```java
package Ques4b;

public class MainApp {

	public static void main(String[] args) {
		// TODO Auto-generated method stub
		MobileUI m= new MobileUI();

		Oven mainOven = new Oven("mainOven");
		GarageGate garageGate = new GarageGate("garageGate");

		Command c1 = new OvenOn(mainOven);
		Command c2 = new OvenOff(mainOven);
		Command c3 = new GarageGateOpen(garageGate);
		Command c4 = new GarageGateClose(garageGate);

		m.setCommand(1, c1);
		m.setCommand(2, c2);
		m.setCommand(3, c3);
		m.setCommand(4, c4);

		m.commmandPressed(1);
		m.commmandPressed(2);
		m.commmandPressed(3);
		m.commmandPressed(4);

	}

}
```