```python
from flask import Flask, jsonify
from flask_cors import cross_origin
app = Flask(__name__)


# home resource
@app.route('/')
@cross_origin()
def home():
    return jsonify({"resource": "home"})

# profile resource
@app.route('/profile')
def profile():
    return jsonify({"resource": "profile"})


if __name__ == "__main__":
    app.run(debug=True)
```

Suppose the API is running on 'http://localhost:5000" 'Which of the following is correct regarding the fetch call to the routes '/' and '/profile' (assuming the client to be Chrome browser)?

Options :

A. ✖ A fetch call to get the home resource from origin "http://localhost:8080" will result in CORS error.

B. ✔ A fetch call to get the home resource from origin "http://localhost:8080" will be successful.

C. ✔ A fetch call to get the profile resource from origin "http://localhost:8080" will result in CORS error.

D. ✖ A fetch call to get the profile resource from origin "http://localhost:8080" will be successful.

Java

Number of Questions : 16

Section Marks : 50

Question Number : 159 Question Type : MCQ

Correct Marks : 0

Question Label : Multiple Choice Question

THIS IS QUESTION PAPER FOR THE SUBJECT "DIPLOMA LEVEL: JAVA "

ARE YOU SURE YOU HAVE TO WRITE EXAM FOR THIS SUBJECT?
CROSS CHECK YOUR HALL TICKET TO CONFIRM THE SUBJECTS TO BE WRITTEN.

(IF IT IS NOT THE CORRECT SUBJECT, PLS CHECK THE SECTION AT THE TOP FOR THE SUBJECTS REGISTERED BY YOU)

Options :

A. ✔ YES

B. ✖ NO

Question Number : 160 Question Type : MCQ

Correct Marks : 2

Question Label : Multiple Choice Question

Consider the following code and choose the correct option.

```
public class DivisionException extends Exception {
  public DivisionException(String msg) {
    super(msg);
  }
}
public class Test {
  public static void divideby7() throws DivisionException {
    throw new DivisionException("Division by 7 is not allowed");
  }
  public static void main(String[] args) {
    try {
      divideby7();
      System.out.print("Completed successfully");    // LINE-1
    }
    catch(DivisionException de) {                    // LINE-2
    }
    catch(Exception e) {                             // LINE-3
    }
  }
}
```

Options :

A. ✖ Compilation fails because the print statement in LINE-1 is not reachable.

B. ✔ Compilation succeeds

C. ✖ Compilation fails because the exception handler in LINE-3 is not reachable.

D. ✖ The code generates the output:

Completed successfully

Question Number : 161 Question Type : MCQ

Correct Marks : 2

Question Label : Multiple Choice Question

Consider the following code.

```
import java.util.*;
public class Test{
  public static void main(String[] args) {
    var m = new HashMap<String, Integer>();
    m.put("Raju", 1991);
    m.put("Mohan", 1995);
    m.put("Shiva", 1996);
    m.put("Vinay", 1993);
    m.put("Mahi", 1992);
    int year = 0;
    String name = "";
    Set<String> keys = m.keySet();
    for(String s : keys){
      int e =  m.get(s);
      if(e > year){
        year = e;
        name = s;
      }
    }
    System.out.println(name);
  }
}
```

What will the output be?

Options :

A. ✔ Shiva

B. ✖ Mahi

C. ✖ Raju

D. ✖ Compilation error

Question Number : 162 Question Type : MCQ

Correct Marks : 2

Question Label : Multiple Choice Question

Consider the following code.

```
import java.util.*;
public class Test {
  public static void main(String[] args) {
    List<Integer> runs1 = new ArrayList<>();
    runs1.add(34);
    runs1.add(42);
    runs1.add(50);

    List<Integer> runs2 = new ArrayList<>();
    runs2.add(45);
    runs2.add(90);
    runs2.add(34);

    Map<String, Integer> am = new HashMap<>();
    Map<String, List<Integer>> hm = new HashMap<>();
    hm.put("Anil", runs1);
    hm.put("Vikas", runs2);
    ***----------------***
         CODE BLOCK
    ***----------------***
  }
}
```

Choose the correct option to fill in the CODE BLOCK that adds the name and the average runs of both the players as map entries in Map<String, Integer> am.

Options :

A. ✖
```
Set<String> names = hm.keySet();
for(Set name : names){
    List<Integer> temp = hm.get(name);
    int count = 0;
    int sum = 0;
    for(List i : temp){
        count = count + 1;
        sum = sum + i;
    }
    int avg = sum/count;
    am.put(name, avg);
}
```

B. ✖

```
Set<String> names = hm.keySet();
for(Set<String> name : names){
    List<Integer> temp = hm.get(name);
    int count = 0;
    int sum = 0;
    for(List<Integer> i : temp){
        count = count + 1;
        sum = sum + i;
    }
    int avg = sum/count;
    am.put(name, avg);
}
```

C. ✖

```
    Set<String> names = hm.keySet();
    int count = 0;
    int sum = 0;
    for(String name : names){
        List<Integer> temp = hm.get(name);
        sum = sum + temp;
        count = count + 1;
        int avg = sum/count;
        am.put(name, avg);
    }
```

D. ✔

```
    Set<String> names = hm.keySet();
    for(String name : names){
        List<Integer> temp = hm.get(name);
        int count = 0;
        int sum = 0;
        for(Integer i : temp){
            count = count + 1;
            sum = sum + i;
        }
        int avg = sum/count;
        am.put(name, avg);
    }
```

Question Number : 163 Question Type : MCQ

Correct Marks : 3

Question Label : Multiple Choice Question

Consider two Java files located in two different packages as shown below.

```
//MathUtility.java
package mathutil;
public class MathUtility{
    public long getFactorial(int n) { // returns n!  }
    protected long add(long n1,long n2) { // returns n1+n2 }
    long subtract(long n1, long n2) { // return n1-n2 }
}

//Test1.java
package test;
import mathutil.*;
class Calculator extends MathUtility{
    public long FactorialSum(int num, int i) {
        long fact = this.getFactorial(i);        // LINE-1
        long sum = this.add(num, fact);          // LINE-2
        return sum;
    }
    public long FactorialDifference(int num, int i) {
        long fact = this.getFactorial(i);        // LINE-3
        long diff = this.subtract(num, fact);    // LINE-4
        return diff;
    }
}

public class Test{
    public static void main(String args[]) {
        Calculator c = new Calculator();
        System.out.println("Factorial Sum: "+ c.FactorialSum(4,2));
        System.out.println("Factorial Difference: "+ c.FactorialDifference(4,2));
    }
}
```

Choose the correct option regarding these two .java files.

Options :

A. ✖ LINE-1 & LINE-3 will lead to compilation error.

B. ✖ LINE-2 will lead to compilation error.

C. ✖ LINE-3 will lead to compilation error.

D. ✔ LINE-4 will lead to compilation error.

Question Number : 164 Question Type : MCQ

Correct Marks : 3

Question Label : Multiple Choice Question

Consider the following code.

```
class ArrayOperations{
  public <T> void display(T[] arr){
    for(T e : arr)
        //print e
  }
  public <T extends Number> T findMax(T[] elements){
    T max = elements[0];
    // returns the largest element in the array
  }
}
```

What is class ArrayOperations converted to, after type erasure?

Options :

A. ✖
```
class ArrayOperations{
  public void display(T[] arr){
    for(T e : arr)
      //print e
  }
  public Number findMax(Number[] elements){
    Number max = elements[0];
    // returns the largest element in the array
  }
}
```

B. ✖
```
class ArrayOperations{
  public void display(Object[] arr){
    for(Object e : arr)
     //print e
  }
  public Double findMax(Double[] elements){
    Double max = elements[0];
    // returns the largest element in the array
  }
}
```

C. ✔

```
class ArrayOperations{
  public void display(Object[] arr){
    for(Object e : arr)
      //print e
  }
  public Number findMax(Number[] elements){
    Number max = elements[0];
    // returns the largest element in the array
  }
  }
```

D. ✖

```
class ArrayOperations{
  public void display(Object[] arr){
    for(Object e : arr)
      //print e
  }
  public Object findMax(Object[] elements){
    Object max = elements[0];
    // returns the largest element in the array
  }
}
```

Question Number : 165 Question Type : MCQ

Correct Marks : 3

Question Label : Multiple Choice Question

Consider the following code.

```
import java.util.*;
interface Shape{
  public abstract int area();
}
class Rectangle implements Shape, Cloneable{
  int length, breadth;
  public Rectangle(int l, int b){
    length = l;
    breadth = b;
  }
  public int area(){
    return length * breadth;
  }
  public Rectangle clone() throws CloneNotSupportedException{
    return (Rectangle)super.clone();
  }
}
public class Test{
  public static void main(String[] args){
    Rectangle r1 = new Rectangle(10, 2);
    try{
      Rectangle r2 = r1.clone();
      r1.length = 4;
      r2.breadth = 3;
      System.out.print(r1.area() + r2.area());
    }
    catch(CloneNotSupportedException e){
      System.out.println("Cloning not supported");
    }
  }
}
```

What will the output be?

Options :

A. ✔ 38

B. ✖ 40

C. ✖ 24

D. ✖ 28

Question Number : 166 Question Type : MCQ

Correct Marks : 3

Question Label : Multiple Choice Question

Consider the code given below.

```
import java.util.*;
class Calculator{
    public static int getMax(int a, int b){
        if(a > b)
            return a;
        return b;
    }
}
public class FClass{
    public static void main(String[] args){
        Map<String, Integer> asgmt1 = new TreeMap<String, Integer>();
        Map<String, Integer> asgmt2 = new TreeMap<String, Integer>();
        Map<String, Integer> score = new TreeMap<String, Integer>();

        asgmt1.put("math", 66);
        asgmt1.put("physics", 76);
        asgmt1.put("chemistry", 66);

        asgmt2.put("biology", 76);
        asgmt2.put("physics", 56);
        asgmt2.put("chemistry", 76);

         for (Map.Entry<String, Integer> e : asgmt1.entrySet())
             score.put(e.getKey(), e.getValue());

         for (Map.Entry<String, Integer> e : asgmt2.entrySet())
             score.merge(e.getKey(), e.getValue(), Calculator::getMax);

         for (Map.Entry<String, Integer> e : score.entrySet())
             System.out.print(e.getKey() + " : " + e.getValue() + ", ");
     }
 }
```

What will the output be?

Options :

A. ✖ biology : 76, chemistry : 66, math : 66, physics : 76,

B. ✖ biology : 76, chemistry : 76, math : 66, physics : 56,

C. ✔ biology : 76, chemistry : 76, math : 66, physics : 76,

D. ✖ biology : 76, chemistry : 66, chemistry : 76, math : 66, physics : 76, physics : 56,

Question Number : 167 Question Type : MCQ

Correct Marks : 3

Question Label : Multiple Choice Question

Consider the code given below.

```
class Address{
    private String addr;
    private long pincode;
    public Address(String a, long p){
        assert a.length() == 4: "addr = " + a;  //assert-1
        addr = a;
        assert p >= 100000 && p <= 999999: "pincode = " + p;   //assert-2
        pincode = p;
    }
}
class Product{
    private String name;
    public Product(String n){
        assert n.length() == 4: "pname = " + n;    //assert-3
        name = n;
    }
}
class Order{
    private Product pd;
    private int qty;
    private Address shpAddr;
    public Order(String pn, int q, String addr, long pc){
        pd = new Product(pn);
        assert q > 0: "qty = " + q;  //assert-4
        qty = q;
        shpAddr = new Address(addr, pc);
    }
}
public class FClass{
    public static void main(String[] args){
        Order od = new Order("P12", 10, "DELH", 65234);
    }
}
```

Identify the `assert` statement that throws the `AssertionError` when the class is executed as:

```
java -ea:...  -da:Product FClass
```

Options :

A. ✖ assert-1

B. ✔ assert-2

C. ✖ assert-3

D. ✖ assert-4

Question Number : 168 Question Type : MCQ

Correct Marks : 3

Question Label : Multiple Choice Question

The following Java program displays customized messages about three students, based on their credits and attendance for a course. Consider this code, and answer the question that follows.

```
class CreditsException extends Exception{
    public CreditsException(String str){
        super(str);
    }
}
class AttendanceException extends Exception{
    public AttendanceException(String str){
        super(str);
    }
}
class Student{
    private String name;
    private int credits;
    private double attendance;
    public Student(String name, int credits,double attendance) {
        this.name = name;
        this.credits = credits;
        this.attendance=attendance;
    }

    public void promote() {
        System.out.print(this.name+" ");
        if(this.credits<40) {
            try {
                throw new CreditsException("has credits shortage.");
            }
            catch (CreditsException e) {
                System.out.println(e.getMessage());
            }
        }
        else {
            if(this.attendance<65) {
                try {
                    throw new AttendanceException("has attendance shortage.");
                }
                catch(AttendanceException e) {
                    System.out.println(e.getMessage());
                }
            }
            else {
                System.out.println("is promoted to final year.");
            }
        }
    }
}
public class ExceptionTest{
    public static void main(String[] args) {
        Student student1=new Student("Virat", 34, 54);
        student1.promote();
        Student student2=new Student("Rohit", 40, 75);
        student2.promote();
        Student student3=new Student("Shivam", 42, 64.5);
        student3.promote();
    }
}
```

Choose the correct option.

Options :

A. ✔ This program generates the output:

Virat has credits shortage.
Rohit is promoted to final year.
Shivam has attendance shortage.

B. ✖ This program generates a compilation error because you cannot create more than one user defined exception in a program.

C. ✖ This program terminates abnormally due to unhandled exception(s).

D. ✖ This program generates the output:

```
Virat has credits shortage.
has attendance shortage.
Rohit is promoted to final year.
Shivam has attendance shortage.
```

Question Number : 169 Question Type : MSQ

Correct Marks : 3

Question Label : Multiple Select Question

Consider the code given below, that prints the maximum area of a set of rectangles, and the maximum area of a set of triangles. Based on the code, answer the question that follows.

```
abstract class Shape{
    public abstract double area();
}
class Rectangle extends Shape{
    // instance variables and constructors here
    // Definition of public double area()
}
class Triangle extends Shape{
    // instance variables and constructors here
    // Definition of public double area()
}
public class Test {
    //LINE-1
    {
        double max = arr[0].area();
        for(int i = 0; i < arr.length; i++){
            if(arr[i].area() > max){
                max = arr[i].area();
            }
        }
        return max;
    }
    public static void main(String[] args){
        Rectangle[] r = {/* 5 Rectangle objects here */ };
        Triangle[] t = {/* 5 Triangle objects here */ };
        double max_rect = maxArea(r);
        double max_trian = maxArea(t);
        System.out.println("Largest area in rectangles: "+max_rect);
        System.out.println("Largest area in triangles: "+max_trian);
    }
}
```

Identify the header(s) for function maxArea (at LINE-1) such that the code prints the largest area in the rectangles and that in the triangles.

Options :

A. ✔ `public static <T extends Shape> double maxArea(T[] arr)`

B. ✖ `public static <T> double maxArea(T[] arr)`

C. ✖ `public static double maxArea(T[] arr)`

D. ✔ `public static double maxArea(Shape[] arr)`

Question Number : 170 Question Type : MSQ

Correct Marks : 3

Question Label : Multiple Select Question

The following Java code maps a set of countries to their capitals. Based on the code, answer the question that follows.

```
import java.util.*;
public class Test {
    public static void main(String[] args) {
        TreeSet<String> set1=new TreeSet<String>();
        HashSet<String> set2=new HashSet<String>();
        HashMap<String, String> map1=new HashMap<String, String>();
        map1.put("India", "New Delhi");
        map1.put("Australia", "Sydney");
        map1.put("Srilanka", "Colombo");
        map1.put("Bangladesh", "Dhaka");

        for(Map.Entry<String, String> entry:map1.entrySet()) {
            set1.add(entry.getKey());
            set2.add(entry.getValue());
        }
        System.out.println(set1);       //LINE-1
        System.out.println(set2);       //LINE-2
    }
}
```

Choose the correct option(s).

Options :

A. ✔ In LINE-1, the values in `set1` are printed in sorted order.

B. ✖ In LINE-2, the values in `set2` are printed in sorted order.

C. ✖ We cannot predict the order in which elements of `set1` are printed.

D. ✔ We cannot predict the order in which elements of `set2` are printed.

Question Number : 171 Question Type : MCQ

Correct Marks : 5

Question Label : Multiple Choice Question

Consider the code given below.

```
import java.util.*;
class School implements Cloneable{
    String schoolName;
    String place;
    public School(String s, String p){
        schoolName = s;
        place  = p;
    }
    public School clone() throws CloneNotSupportedException{
        return (School)super.clone();
    }
}
class Student implements Cloneable{
    School sch;
    String stuName;
    public Student(String n, School s) {
        sch = s;
        stuName = n;
    }
    public Student clone() throws CloneNotSupportedException{
        Student s = (Student) super.clone();
        return s;
    }
    public String toString(){
        return stuName + ":"+ sch.schoolName + ":" + sch.place;
    }
}

public class Test{
    public static void main(String[] args){
        Student s1 = new Student("Vasanth", new School("DAV", "Chennai"));
        try{
            Student s2 = s1.clone();
            s2.stuName = "Neeraj";
            s2.sch.place = "Delhi";
            System.out.println(s1 + ", " + s2);
        }
        catch(CloneNotSupportedException e){
            System.out.println("Cloning not supported");
        }
    }
}
```

What will the output be?

Options :

A. ✖ Cloning not supported

B. ✔ Vasanth:DAV:Delhi, Neeraj:DAV:Delhi

C. ✖ Vasanth:DAV:Chennai, Neeraj:DAV:Delhi

D. ✖ Neeraj:DAV:Delhi, Neeraj:DAV:Delhi

Question Number : 172 Question Type : MCQ

Correct Marks : 5

Question Label : Multiple Choice Question

The following code maps a set of names of students to their scores, and groups the names based on their scores. Based on the code, answer the question that follows.

```
import java.util.*;
public class MapTest {
    TreeSet<String> set1=new TreeSet<String>();
    TreeSet<String> set2=new TreeSet<String>();
    public boolean property(Integer marks) {
        if(marks>=40)
            return false;
        return true;
    }
    public void validate(TreeMap<String, Integer> stuMap) {
        for(Map.Entry<String, Integer> entry:stuMap.entrySet()) {
            if(property(entry.getValue()))
                set1.add(entry.getKey());
            else
                set2.add(entry.getKey());
        }
    }
    public void display() {
        System.out.println(set1);
        System.out.println(set2);
    }
    public static void main(String[] args) {
        TreeMap<String,Integer> student=new TreeMap<String, Integer>();
        student.put("Beas", 90);
        student.put("Mahanadhi", 36);
        student.put("Kaveri", 80);
        student.put("Krishna", 26);
        student.put("Chambal", 38);
        student.put("Yamuna", 48);
        MapTest obj=new MapTest();
        obj.validate(student);
        obj.display();
    }
}
```

Choose the correct option.

Options :

A. ✖ This program generates the output:
[Beas, Kaveri, Yamuna]
[Chambal, Krishna, Mahanadhi]

B. ✔ This program generates the output:
[Chambal, Krishna, Mahanadhi]
[Beas, Kaveri, Yamuna]

This program generates the output:

```
[Mahanadhi, Krishna, Chambal]
[Beas, Kaveri, Yamuna]
```

C. ✖

D. ✖ We cannot predict the order in which elements of `set1` and `set2` are printed.

Question Number : 173 Question Type : MSQ

Correct Marks : 5

Question Label : Multiple Select Question

Consider the following program.

```
import java.util.*;
class OrderedPair<T extends Number>{
    private T a;
    private T b;
    public OrderedPair(T a_val, T b_val){
        a = a_val;
        b = b_val;
    }
    public T get_a() {
        return a;
    }
    public T get_b() {
        return b;
    }
    public boolean isMinPair(_____ LINE-1 _______) {
        if(this.a.doubleValue() < pair2.get_a().doubleValue()
            && this.b.doubleValue() < pair2.get_b().doubleValue()){

            return true;
        }
        else
            return false;
    }
}

public class Test{
    public static void main(String args[]) {
        OrderedPair<Integer> op1 = new OrderedPair<Integer>(5,6);
        OrderedPair<Integer> op2 = new OrderedPair<Integer>(3,2);

        op2.isMinPair(op1);
    }
}
```

Choose all the options which can be used in place of LINE-1 for successful compilation.

Options :

A. ✖ OrderedPair <Number> pair2

B. ✔ OrderedPair <T> pair2

C. ✖ OrderedPair <Object> pair2

D. ✔ OrderedPair <?> pair2

Question Number : 174 Question Type : MSQ

Correct Marks : 5

Question Label : Multiple Select Question

From among the options, choose the code segment(s) that give(s) the same output as is given by the Java code inside the CODE BLOCK.

```
import java.util.*;
import java.util.stream.*;

public class FClass{
    public static void main(String[] args){
        //CODE BLOCK begins here
        double c = Stream.iterate(1, n -> n+1)
                .map(n -> n * n * n)
                .limit(5)
                .count();
        //CODE BLOCK ends here
        System.out.println(c);
    }
}
```

Options :

A. ✖
```
double c = Stream.iterate(1, n -> n+1)
            .map(n -> n * n * n)
            .filter(n -> n <= 5)
            .count();
```

B. ✔
```
double c = Stream.iterate(1, n -> n <= 5,n -> n + 1)
            .map(n -> n * n * n)
            .limit(5)
            .count();
```

C. ✔
```
double c = Stream.iterate(1, n -> n+1)
            .takeWhile(n -> n <= 5)
            .map(n -> n * n * n)
            .count();
```

```
double c = Stream.iterate(1, n -> n+1)
            .map(n -> n * n * n)
            .dropWhile(n -> n <= 125)
            .count();
```

D. ✖

PDSA

Number of Questions : 14

Section Marks : 50

Question Number : 175 Question Type : MCQ

Correct Marks : 0

Question Label : Multiple Choice Question

THIS IS QUESTION PAPER FOR THE SUBJECT "DIPLOMA LEVEL: PROGRAMMING DATA STRUCTURES AND ALGORITHMS USING PYTHON"

ARE YOU SURE YOU HAVE TO WRITE EXAM FOR THIS SUBJECT?
CROSS CHECK YOUR HALL TICKET TO CONFIRM THE SUBJECTS TO BE WRITTEN.

(IF IT IS NOT THE CORRECT SUBJECT, PLS CHECK THE SECTION AT THE TOP FOR THE SUBJECTS REGISTERED BY YOU)

Options :

A. ✔ YES

B. ✖ NO

Question Number : 176 Question Type : MCQ

Correct Marks : 3

Question Label : Multiple Choice Question

Let *G* be a complete undirected graph on 4 vertices, having 6 edges with weights being 1, 2, 3, 4, 5,