

Object Oriented Programming Mid Exam



From:

AL AZHAR RIZQI RIFA'I FIRDAUS

Class:

2 I

Absence:

01

Student Number Identity:

2241720263

Department:

Information Technology

Study Program:

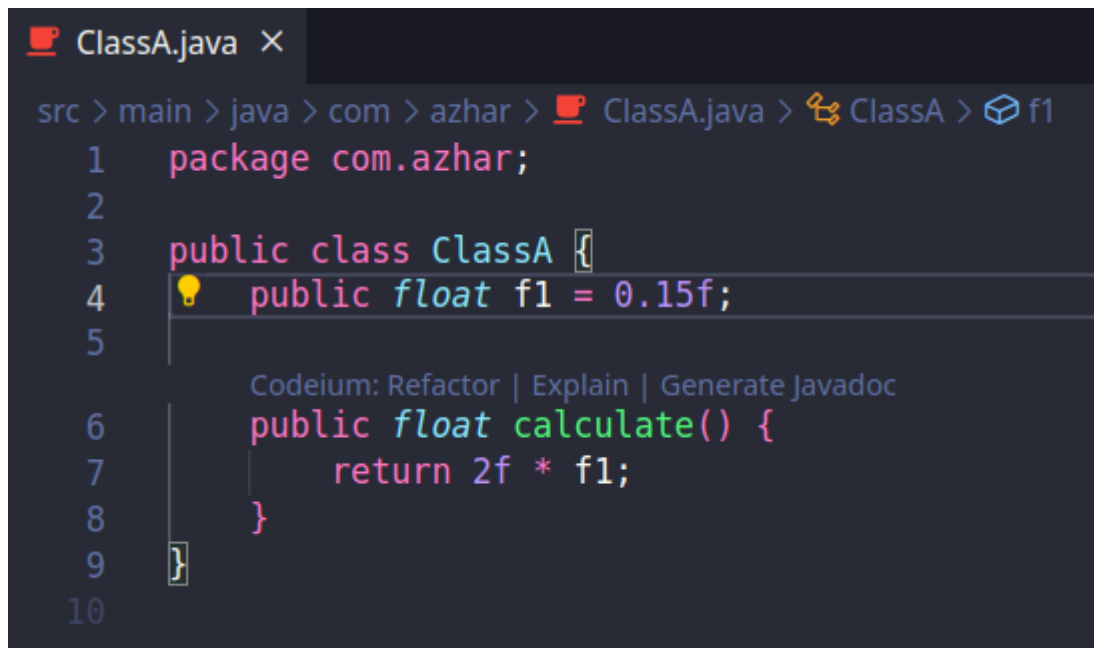
Informatics Engineering

Problem 1: Class Writing

Based on the example class ClassA below, explain whether the source code writing in the example class is correct. If not, what needs to be corrected?

```
public class ClassA {  
    float f1 = 0.15f;  
  
    float hitung() {  
        float x = 2f * f1;  
    }  
}
```

- From code above, we usually used access modifier to create attribute or method. But on code above, there is no access modifier. It make default package private, it means attribute can be access with all of class within one package, but not for outside package. Then at method calculate, there is no return because it not using void. It should be we used access modifier for our attribute and method, then give return for method calculate like below:



```
src > main > java > com > azhar > ClassA.java > ClassA > f1  
1  package com.azhar;  
2  
3  public class ClassA {  
4      public float f1 = 0.15f;  
5  
6      public float calculate() {  
7          return 2f * f1;  
8      }  
9  }  
10
```

Problem 2: Calculation of the Number of Elements of a 2-Dimensional Array

In class SoalArray1, there is a 2-dimensional array with size 3x3. Write the Java code to calculate the total number of array elements using looping.

```
public class SoalArray1 {  
    public static void main(String[] args) {  
        int[][] arrayInt = {{1, 1, 4}, {2, 1, 2}, {3, 2, 1}};  
        // hitung jumlah elemen array 2 dimensi  
        // gunakan perulangan  
    }  
}
```

```
ArrayTask.java ×
src > main > java > com > azhar > ArrayTask.java > ArrayTask > main(String[])
1 package com.azhar;
2
3 public class ArrayTask {
4     Run | Debug | Codeium: Refactor | Explain | Generate Javadoc
5     public static void main(String[] args) {
6         int[][] arrayInt = {{1, 1, 4}, {2, 1, 2}, {3, 2, 1}};
7         // hitung jumlah elemen array 2 dimensi
8         // gunakan perulangan
9         int total = 0;
10        for (int i = 0; i < arrayInt.length; i++) {
11            for (int j = 0; j < arrayInt[i].length; j++) {
12                total += arrayInt[i][j];
13            }
14        }
15        System.out.println("Total amount of elements in array: " + total);
16    }
17 }
18

→ zharsuke@box ~/Documents/College/Semester_3/oop/mid-exam/coding git:(master) ×
InExceptionMessages -cp /home/zharsuke/Documents/College/Semester_3/oop/mid-exam/c
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
Total amount of elements in array: 17
→ zharsuke@box ~/Documents/College/Semester_3/oop/mid-exam/coding git:(master) ×
```

Problem 3: Inheritance of Attributes and Methods

In the given source code, class ClassY is derived from class Class. Name the attributes and methods inherited by ClassY from its parent class (class Class). Also explain what output of the code written in ClassY and how the value is obtained.

```
public class Class {
    int a = 2;
    int x = 0;

    int hitung() {
        x = x + 5 * a;
        return x;
    }
}

public class ClassY extends Class {
    int b = 5;

    int y = 0;

    int hitungY() {
        y = hitung() * b;
        return y;
    }

    public static void main(String[] args) {
        ClassY cy = new ClassY();
        System.out.println(cy.hitungY());
    }
}
```

```
ClassA.java ArrayTask.java Class.java X ClassY.java
src > main > java > com > azhar > Class.java > ...
1 package com.azhar;
2
3 public class Class {
4     int a = 2;
5     int x = 0;
6     Codeium: Refactor | Explain | Generate Javadoc
7     int hitung() {
8         x = x + 5 * a;
9         return x;
10    }
11
```

```
ClassA.java ArrayTask.java Class.java ClassY.java X
src > main > java > com > azhar > ClassY.java > ClassY > main(String[])
1 package com.azhar;
2
3 public class ClassY extends Class {
4     int b = 5;
5     int y = 0;
6     Codeium: Refactor | Explain | Generate Javadoc
7     int hitungY() {
8         y = hitung() * b;
9         return y;
10    }
11
12 Run | Debug | Codeium: Refactor | Explain | Generate Javadoc
13 public static void main(String[] args) {
14     ClassY cy = new ClassY();
15     System.out.println(cy.hitungY());
16 }
```

```
→ zharsuke@box ~/Documents/College/Semester_3/oop/mid-exam/coding git:(master) x
InExceptionMessages -cp /home/zharsuke/Documents/College/Semester_3/oop/mid-exam/c
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
50
→ zharsuke@box ~/Documents/College/Semester_3/oop/mid-exam/coding git:(master) x
```

- From code above, it make return 50. It happen because at Class there is hitung() method that calculate $x + 5 * a$, where $x = 0$, $a = 2$, that will return 10. Then at

classY, it call Class as parent and doing calculate again with hitungY() method. It call hitung() method that contain value 10, then multiply by b which is 5. It'll make result value 50. That's why the output is 50.

Problem 4: Student Class with Constructor

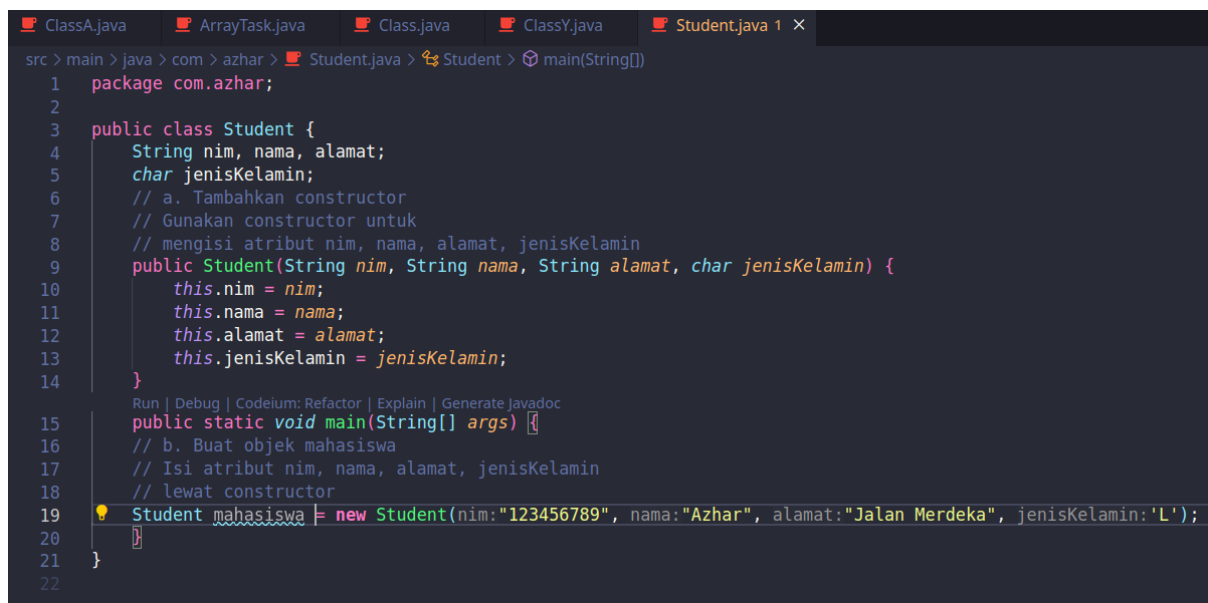
In the Student class, complete the code with:

- Adding a constructor to populate the nim, name, address, and gender attributes.
- Creating a student object and filling the nim, name, address, and gender attributes through the constructor.

```
public class Mahasiswa {
    String nim, nama, alamat;
    char jenisKelamin;

    // a. Tambahkan constructor
    // Gunakan constructor untuk
    // mengisi atribut nim, nama, alamat, jenisKelamin

    public static void main(String[] args) {
        // b. Buat objek mahasiswa
        // Isi atribut nim, nama, alamat, jenisKelamin
        // lewat constructor
    }
}
```

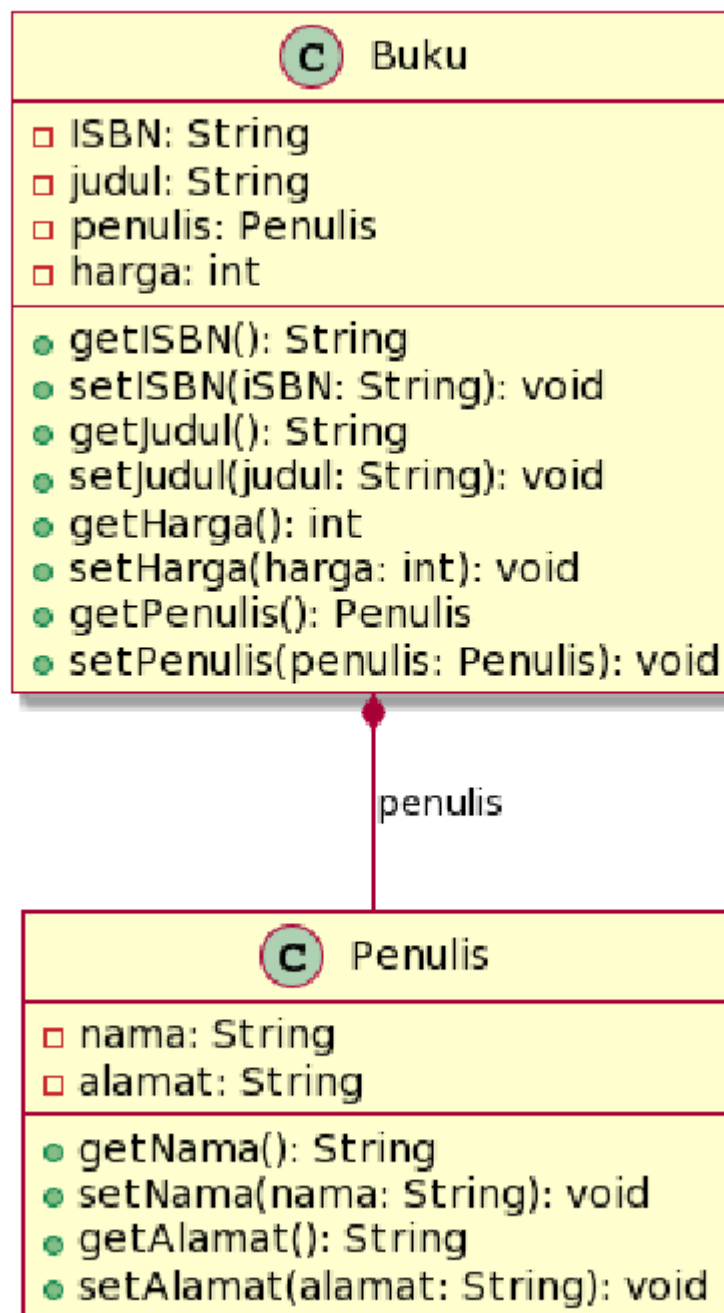


```
src > main > java > com > azhar > Student.java > Student > main(String[])
1  package com.azhar;
2
3  public class Student {
4      String nim, nama, alamat;
5      char jenisKelamin;
6      // a. Tambahkan constructor
7      // Gunakan constructor untuk
8      // mengisi atribut nim, nama, alamat, jenisKelamin
9      public Student(String nim, String nama, String alamat, char jenisKelamin) {
10         this.nim = nim;
11         this.nama = nama;
12         this.alamat = alamat;
13         this.jenisKelamin = jenisKelamin;
14     }
15     Run | Debug | Codeium: Refactor | Explain | Generate Javadoc
16     public static void main(String[] args) {
17         // b. Buat objek mahasiswa
18         // Isi atribut nim, nama, alamat, jenisKelamin
19         // lewat constructor
20         Student mahasiswa = new Student(nim:"123456789", nama:"Azhar", alamat:"Jalan Merdeka", jenisKelamin:'L');
21     }
22 }
```

Problem 5: OOP Book -> Author

Consider the following class diagrams and create source code in java language based on the class diagram.

Class Diagram - Buku and Penulis



Book.java ×

MainBook.java

Author.java

src > main > java > com > azhar > Book.java > Book

```
1  package com.azhar;
2
3  public class Book {
4      private Author author;
5      private String isbn, title;
6      private int price;
7
8      Codeium: Refactor | Explain | Generate Javadoc
9      public String getIsbn() {
10         return isbn;
11     }
12
13     Codeium: Refactor | Explain | Generate Javadoc
14     public String getTitle() {
15         return title;
16     }
17
18     Codeium: Refactor | Explain | Generate Javadoc
19     public int getPrice() {
20         return price;
21     }
22
23     Codeium: Refactor | Explain | Generate Javadoc
24     public Author getAuthor() {
25         return author;
26     }
27
28     Codeium: Refactor | Explain | Generate Javadoc
29     public void setAuthor(Author author) {
30         this.author = author;
31     }
32
33     Codeium: Refactor | Explain | Generate Javadoc
34     public void setIsbn(String isbn) {
35         this.isbn = isbn;
36     }
37
38     Codeium: Refactor | Explain | Generate Javadoc
39     public void setTitle(String title) {
40         this.title = title;
41     }
42 }
```



```
Codeium: Refactor | Explain | Generate Javadoc
36     public void setPrice(int price) {
37         this.price = price;
38     }
39 }
40
```

```
Book.java MainBook.java Author.java X
src > main > java > com > azhar > Author.java > Author
1  package com.azhar;
2
3  public class Author {
4      private String name, address;
5
6      Codeium: Refactor | Explain | Generate Javadoc
7      public String getName() {
8          return name;
9      }
10
11      Codeium: Refactor | Explain | Generate Javadoc
12      public String getAddress() {
13          return address;
14      }
15
16      Codeium: Refactor | Explain | Generate Javadoc
17      public void setName(String name) {
18          this.name = name;
19      }
20
21      Codeium: Refactor | Explain | Generate Javadoc
22      public void setAddress(String address) {
23          this.address = address;
24      }
25  }
```

```
Book.java MainBook.java X Author.java
src > main > java > com > azhar > MainBook.java > MainBook > main(String[])
1 package com.azhar;
2
3 public class MainBook {
    Run | Debug | Codeium: Refactor | Explain | Generate Javadoc
4     public static void main(String[] args) {
5         Author author = new Author();
6         author.setName(name:"Azhar");
7         author.setAddress(address:"Jl. Sukarno Hatta No.9");
8
9         Book book = new Book();
10        book.setAuthor(author);
11        book.setIsbn(isbn:"978-1234567890");
12        book.setTitle(title:"Hacking Hackers");
13        book.setPrice(price:100_000);
14
15        // Accessing author information from the book
16        Author bookAuthor = book.getAuthor();
17        System.out.println("Book Author: " + bookAuthor.getName());
18        System.out.println("Author's Address: " + bookAuthor.getAddress());
19
20        System.out.println("ISBN: " + book.getIsbn());
21        System.out.println("Title: " + book.getTitle());
22        System.out.println("Price: $" + book.getPrice());
23
24    }
25 }
26
```

```
→ zharsuke@box ~/Documents/College/Semester_3/oop/mid-exam/coding git:(master) x
InExceptionMessages -cp /home/zharsuke/Documents/College/Semester_3/oop/mid-exam/c
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
Book Author: Azhar
Author's Address: Jl. Sukarno Hatta No.9
ISBN: 978-1234567890
Title: Hacking Hackers
Price: $100000
→ zharsuke@box ~/Documents/College/Semester_3/oop/mid-exam/coding git:(master) x
```