SOFTWARE ENGINEERING (Class:5215, 5216, 5217 & 5218) - Fall 2021 >
☑ Tests & Quizzes

Tests & Quizzes

Add | Assessments | Question Pools | Event Log | User Activity Report

REYYAN AHMED

Submission Status | Total Scores | Questions | Statistics

Second hourly

Comments for Student:

Table of Contents

- 5 Points 19. During generalization, the private attributes of the base class are not inherited by the child class
- 5 Points 20. Aggregation is a stronger class relationship than composition

Part 2 - Design Related Questions - 0/5 Answered Question, 0.0 / 50.0 Points

- 10 Points 1. Look at the class diagram above carefully. The code below is based on the diagram given above. It however has errors. Can you mark the lines that have an error? As before, a wrong answer will negate a right one
- 10 Points 2. Look at the code below: class MainClass { public static void main(String[] args) { Beta beta; Alpha alpha = new Alpha(); beta = alpha.GetBeta(); alpha = null; } } class Alpha { Beta temp; public Beta GetBeta() { temp = new Beta() return temp; } } class Beta { } Select the correct relationship between the classes
- 10 Points 3. Look at the class diagram above carefully. Then look at the code given below and mark the lines where you find errors. Keep in mind that if you select a wrong option, it will negate a right option that you have selected.
- 10 Points 4. Look at the class diagram above carefully. The code below is based on the diagram given above. Suppose, you fix all the errors, what will be the output of the program?
- 10 Points 5. Look at the class diagram above carefully. Then look at the code given below and mark the lines where you find errors. Keep in mind that if you select a wrong option, it will negate a right option that you have selected.

Part 1 of 2 Default

- Question 1 of 20: 0.0 / 5.0 Points

  During generalization, the protected attributes of the base class are not inherited by the child class

  ✖ ○ True
  ○ False

  Answer Key: False

  Comments for Student:

  Attachments

  No Attachment(s) yet

  Add Attachments

- Question 2 of 20: 0.0 / 5.0 Points

Always use scalar graphics for your icons. It’s the easiest way to ensure your icons will look sharp in any device or resolution

✖ ○ True
○ False

Answer Key: False

Comments for Student:

Attachments

No Attachment(s) yet

Add Attachments

- Question 3 of 20: 0.0 / 5.0 Points

If a base class's reference is pointing towards a child class's object, then it can call the functions defined in the child class

✖ ○ True
○ False

Answer Key: False

Comments for Student:

Attachments

No Attachment(s) yet

Add Attachments

- Question 4 of 20: 0.0 / 5.0 Points

Time runs from left to right in a sequence diagram

✖ ○ True

○ False

Answer Key: False

Comments for Student:

Attachments

No Attachment(s) yet

Add Attachments

- Question 5 of 20: 0.0 / 5.0 Points

The activity final node, drawn as a filled circle, marks the end of the activity

○ True

✖ ○ False

Answer Key: False

Comments for Student:

Attachments

No Attachment(s) yet

Add Attachments

- Question 6 of 20: 5.0 / 5.0 Points

attributes and operations having package visibility are specified using the ______ symbol

- ✔ ○ A. #
- ✔ ○ B. ~
- ✔ ○ C. +

- ✓ ○ D. -

Answer Key: B

Comments for Student:

Attachments

No Attachment(s) yet

Add Attachments

- Question 7 of 20: 0.0 / 5.0 Points

A static attribute in UML is always _________

- ✗ ○ A. Bold
- ✓ ○ B. Italicized
- ✓ ○ C. Underlined

Answer Key: C

Comments for Student:

Attachments

No Attachment(s) yet

Add Attachments

- Question 8 of 20: 0.0 / 5.0 Points

in a class diagram there are _______ ways to define a class

- ✓ ○ A. 2

- ✗ ○ B. 3
- ✓ ○ C. 4
- ✓ ○ D. 5

Answer Key: C

Comments for Student:

Attachments

No Attachment(s) yet

Add Attachments

- Question 9 of 20: 0.0 / 5.0 Points

which of the following diagrams is time oriented

- ✓ ○ A. Use case diagram
- ✓ ○ B. Class diagram
- ✗ ○ C. Activity diagram
- ✓ ○ D. Sequence Diagram

Answer Key: D

Comments for Student:

Attachments

No Attachment(s) yet

Add Attachments

- Question 10 of 20: 5.0 / 5.0 Points

Which of the following diagrams represents the interaction of the user with the software but tells nothing about the internal working of the software?

- ✓ ○ A. Use case diagrams
- ✓ ○ B. Activity diagrams
- ✓ ○ C. Sequence Diagrams
- ✓ ○ D. Class diagrams

Answer Key: A

Comments for Student:

Attachments

No Attachment(s) yet

Add Attachments

- Question 11 of 20: 5.0 / 5.0 Points

In an activity diagram, a decision should always end up at a join

✓

○ True

○ False

Answer Key: False

Comments for Student:

Attachments

No Attachment(s) yet

Add Attachments

- Question 12 of 20: 5.0 / 5.0 Points

While designing software, it is preferable to use composition over generalization

✓
○ True
○ False

Answer Key: True

Comments for Student:

Attachments

No Attachment(s) yet

Add Attachments

- Question 13 of 20: 5.0 / 5.0 Points

A UML diagram that facilitates requirements gathering and interacts between system and external users, is called as

- ✓ ○ A. Activity Diagram
- ✓ ○ B. Class Diagram
- ✓ ○ C. Use Case Diagram
- ✓ ○ D. Sequence Diagram

Answer Key: C

Comments for Student:

Attachments

No Attachment(s) yet

Add Attachments

- Question 14 of 20: 5.0 / 5.0 Points

___________ enables a class to hide the inner details of how it works from the outside world

- ✓ ○ A. Encapsulation
- ✓ ○ B. Abstraction
- ✓ ○ C. Realization
- ✓ ○ D. Generalization

Answer Key: A

Comments for Student:

Attachments

No Attachment(s) yet

Add Attachments

- Question 15 of 20: 0.0 / 5.0 Points

How many views of software can be represented through the Unified Modeling Language (UML)?

- ✓ ○ A. 3
- ✗ ○ B. 4
- ✓ ○ C. 5
- ✓ ○ D. 6

Answer Key: C

Comments for Student:

Attachments

No Attachment(s) yet

Add Attachments

- Question 16 of 20: 5.0 / 5.0 Points

A communication line in a Use case diagram is shown by a ________________

- ✓ ○ A. dotted line
- ✓ ○ B. straight line
- ✓ ○ C. arrow
- ✓ ○ D. dotted arrow

Answer Key: B

Comments for Student:

Attachments

No Attachment(s) yet

Add Attachments

- Question 17 of 20: 5.0 / 5.0 Points

While defining a class, discarding irrelevant details within a given context is called ___________

- ✓ ○ A. Encapsulation
- ✓ ○ B. Abstraction
- ✓ ○ C. Realization
- ✓ ○ D. Generalization

Answer Key: B

Comments for Student:

Attachments

No Attachment(s) yet

Add Attachments

- Question 18 of 20: 0.0 / 5.0 Points

<<extend>> relationship shows that one use case is a special type of another use case

✖ ○ True
○ False

Answer Key: False

Comments for Student:

Attachments

No Attachment(s) yet

Add Attachments

- Question 19 of 20: 0.0 / 5.0 Points

During generalization, the private attributes of the base class are not inherited by the child class

✖ ○ True
○ False

Answer Key: False

Comments for Student:

Attachments

No Attachment(s) yet

Add Attachments

- Question 20 of 20: 0.0 / 5.0 Points

Aggregation is a stronger class relationship than composition

✖ ○ True
○ False

Answer Key: False

Comments for Student:

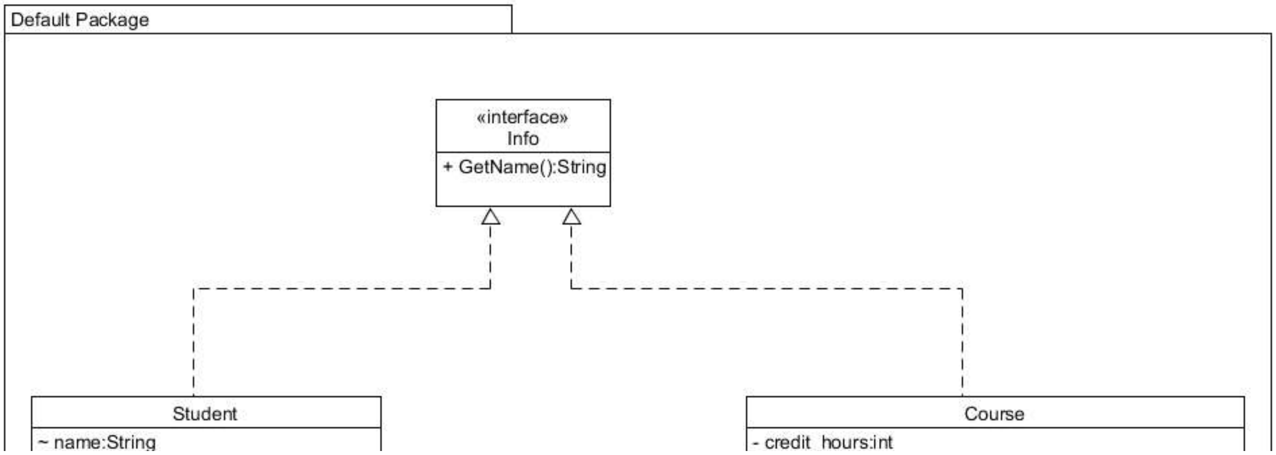
Attachments

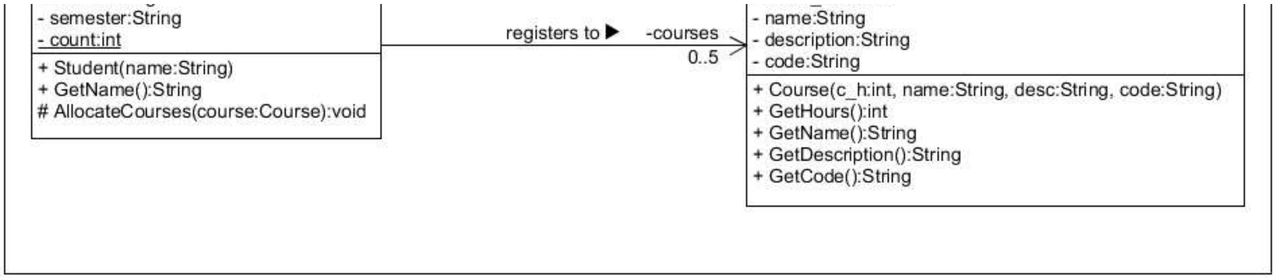
No Attachment(s) yet

Add Attachments

Part 2 of 2 Design Related Questions

- Question 1 of 5: 0.0 / 10.0 Points

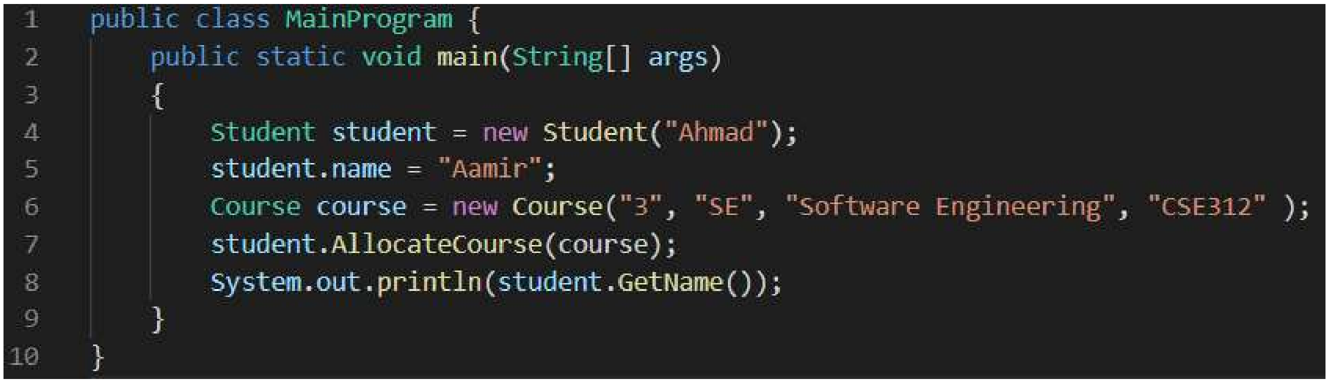




Look at the class diagram above carefully.

The code below is based on the diagram given above. It however has errors. Can you mark the lines that have an error?

As before, a wrong answer will negate a right one



```
1  public class MainProgram {
2      public static void main(String[] args)
3      {
4          Student student = new Student("Ahmad");
5          student.name = "Aamir";
6          Course course = new Course("3", "SE", "Software Engineering", "CSE312" );
7          student.AllocateCourse(course);
8          System.out.println(student.GetName());
9      }
10 }
```

☐ A. 4

☐ B. 5

☐ C. 6

☐ D. 7

☐ E. 8

Answer Key: C

Comments for Student:

Attachments

No Attachment(s) yet

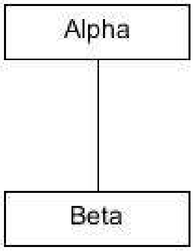
Add Attachments

Question 2 of 5: 0.0 / 10.0 Points

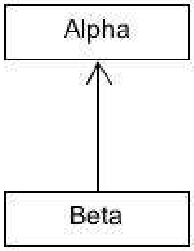
Look at the code below:

```
class MainClass

{

    public static void main(String[] args)

    {

        Beta beta;

        Alpha alpha = new Alpha();

        beta = alpha.GetBeta();

        alpha = null;

    }

}


class Alpha

{

    Beta temp;

    public Beta GetBeta()

    {

        temp = new Beta()

        return temp;

    }

}


class Beta

{

}
```

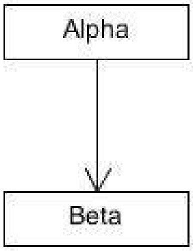
Select the correct relationship between the classes



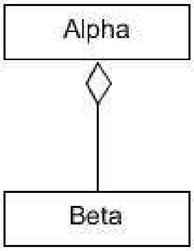
(A)



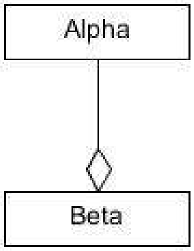
(B)



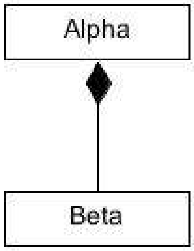
(C)



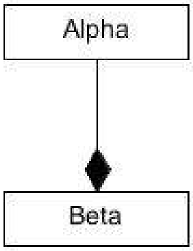
(D)



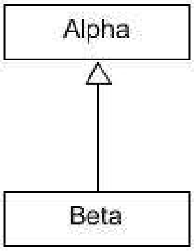
(E)



(F)



(G)



(H)

- ✔ ○ A. A
- ✔ ○ B. B
- ✔ ○ C. C
- ✔ ○ D. D
- ✔ ○ E. E
- ✔ ○ F. F
- ✔ ○ G. G
- ✔ ○ H. H

Answer Key: D

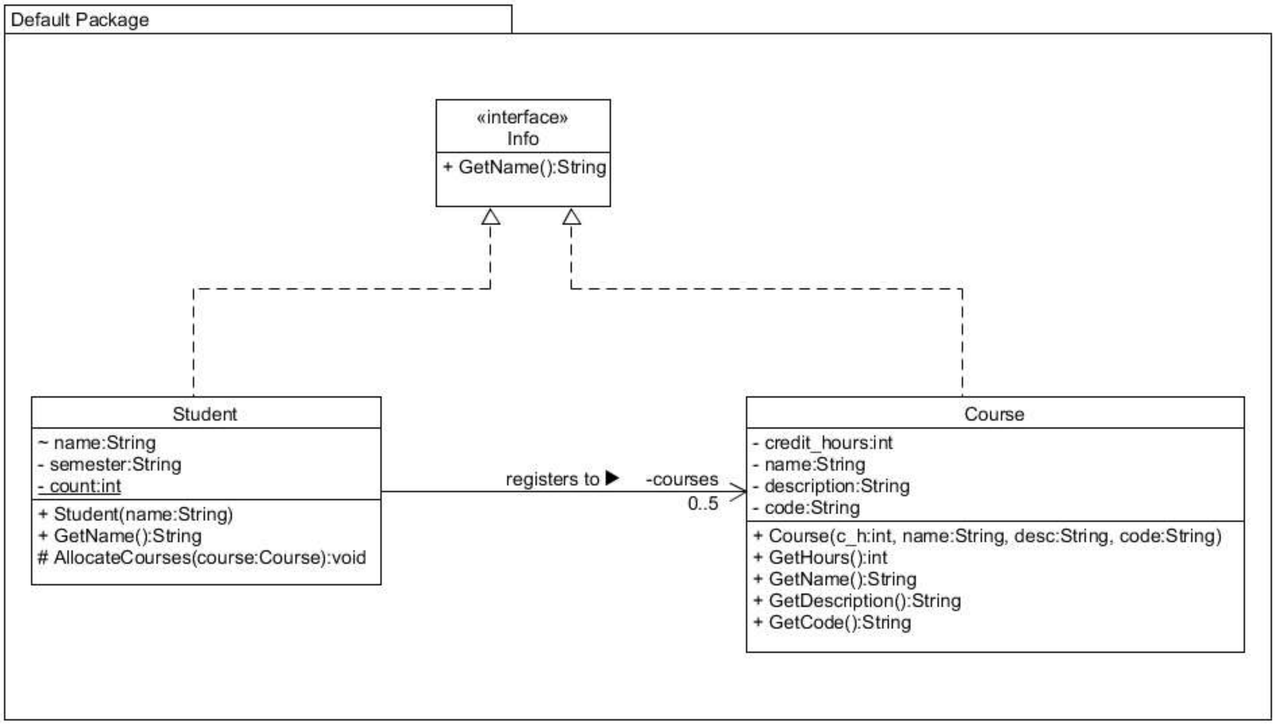
Comments for Student:

Attachments

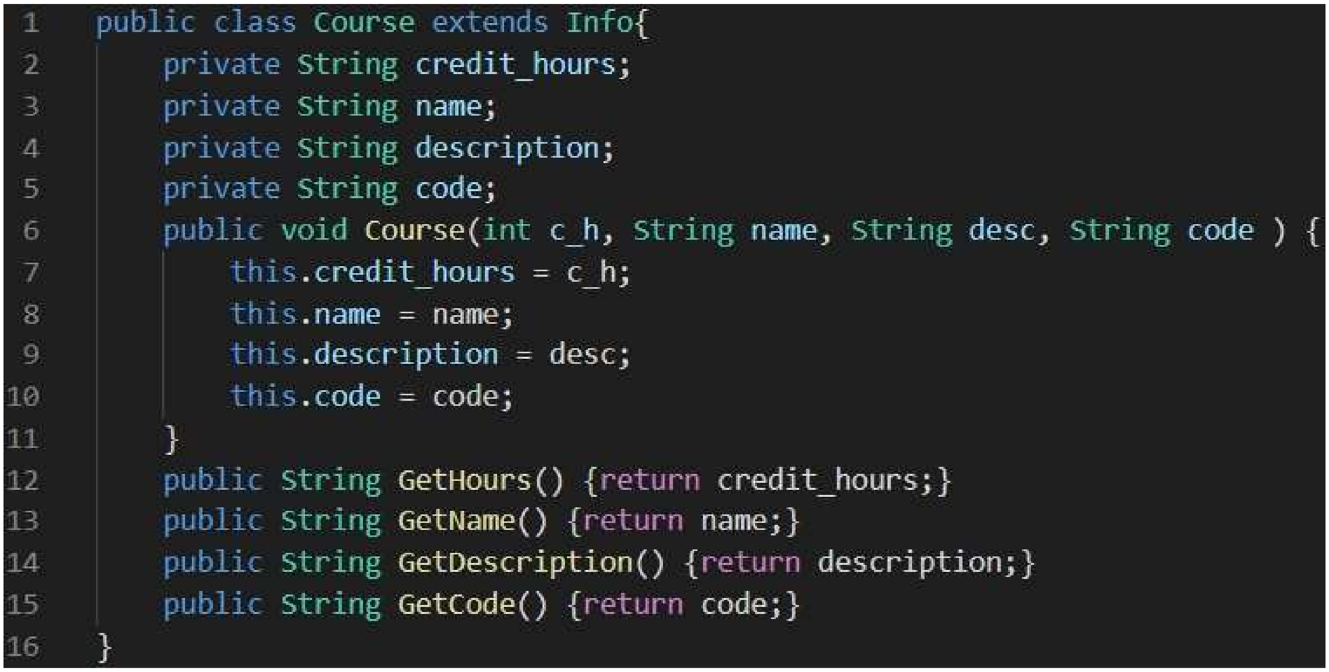
No Attachment(s) yet

Add Attachments

- Question 3 of 5: 0.0 / 10.0 Points



Look at the class diagram above carefully. Then look at the code given below and mark the lines where you find errors. Keep in mind that if you select a wrong option, it will negate a right option that you have selected.



```
1   public class Course extends Info{
2       private String credit_hours;
3       private String name;
4       private String description;
5       private String code;
6       public void Course(int c_h, String name, String desc, String code ) {
7           this.credit_hours = c_h;
8           this.name = name;
9           this.description = desc;
10          this.code = code;
11      }
12      public String GetHours() {return credit_hours;}
13      public String GetName() {return name;}
14      public String GetDescription() {return description;}
15      public String GetCode() {return code;}
16  }
```

□ A. 1

□ B. 2

□ C. 3

□ D. 4

□ E. 5

□ F. 6

□ G. 7

□ H. 8

□ I. 9

□ J. 10

□ K. 11

□ L. 12

□ M. 13

□ N. 14

□ O. 15

□ P. 16

Answer Key: A, B, F, L

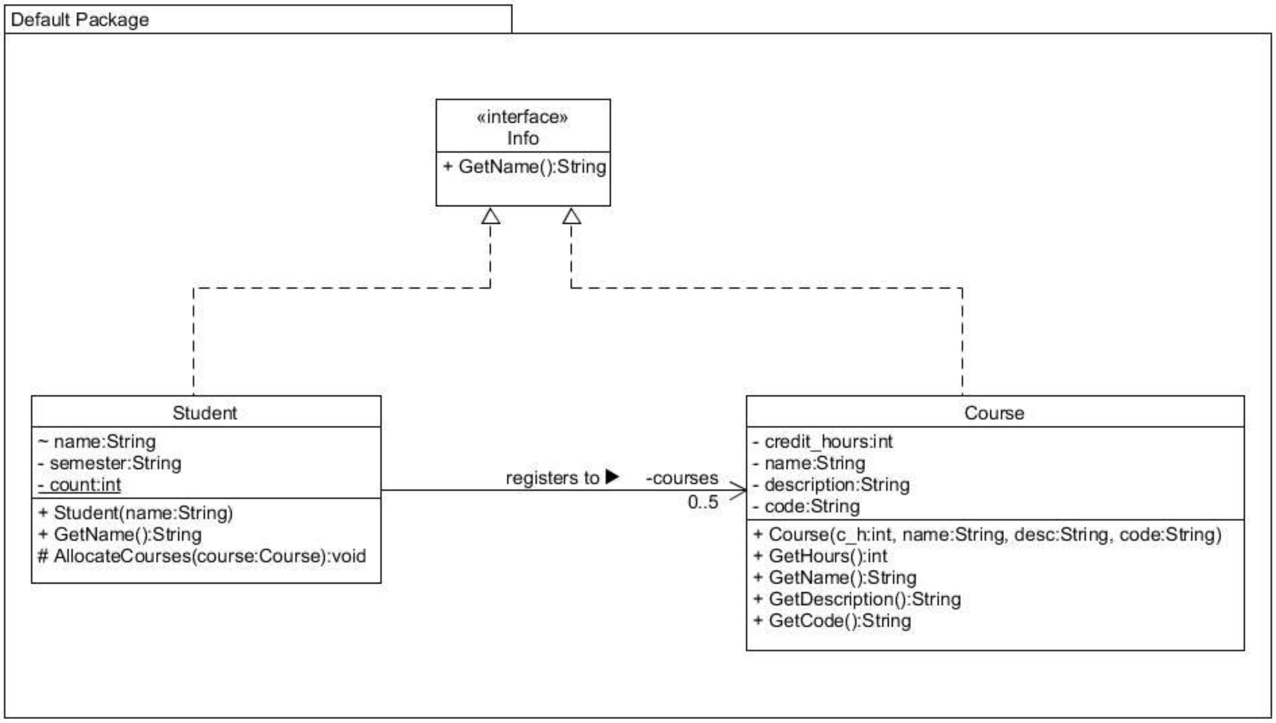
Comments for Student:

Attachments

No Attachment(s) yet

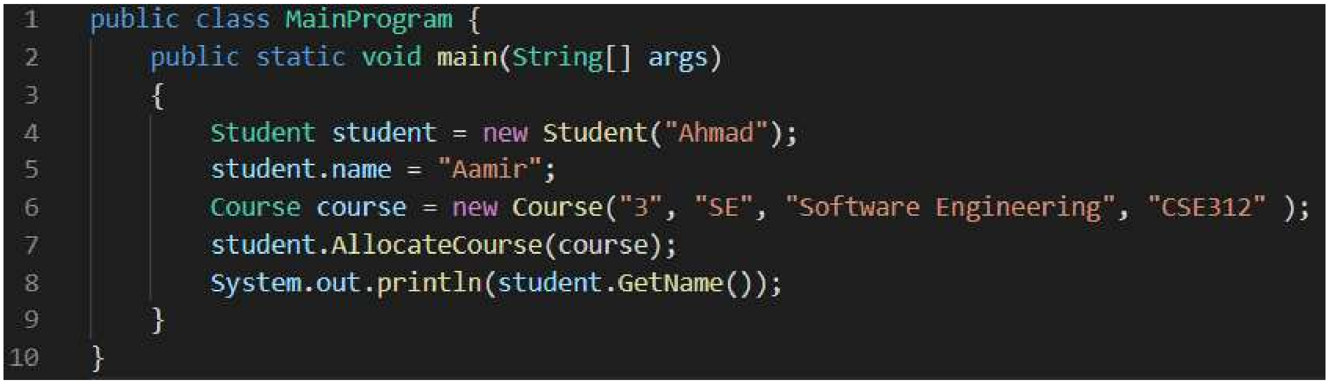
Add Attachments

- Question 4 of 5: 0.0 / 10.0 Points



Look at the class diagram above carefully.

The code below is based on the diagram given above. Suppose, you fix all the errors, what will be the output of the program?



```
public class MainProgram {
    public static void main(String[] args)
    {
        Student student = new Student("Ahmad");
        student.name = "Aamir";
        Course course = new Course("3", "SE", "Software Engineering", "CSE312" );
        student.AllocateCourse(course);
        System.out.println(student.GetName());
    }
}
```

- ✓ ○ A. Ahmad
- ✓ ○ B. Aamir
- ✓ ○ C. null
- ✓ ○ D. The program will exit without showing an error

Answer Key: B

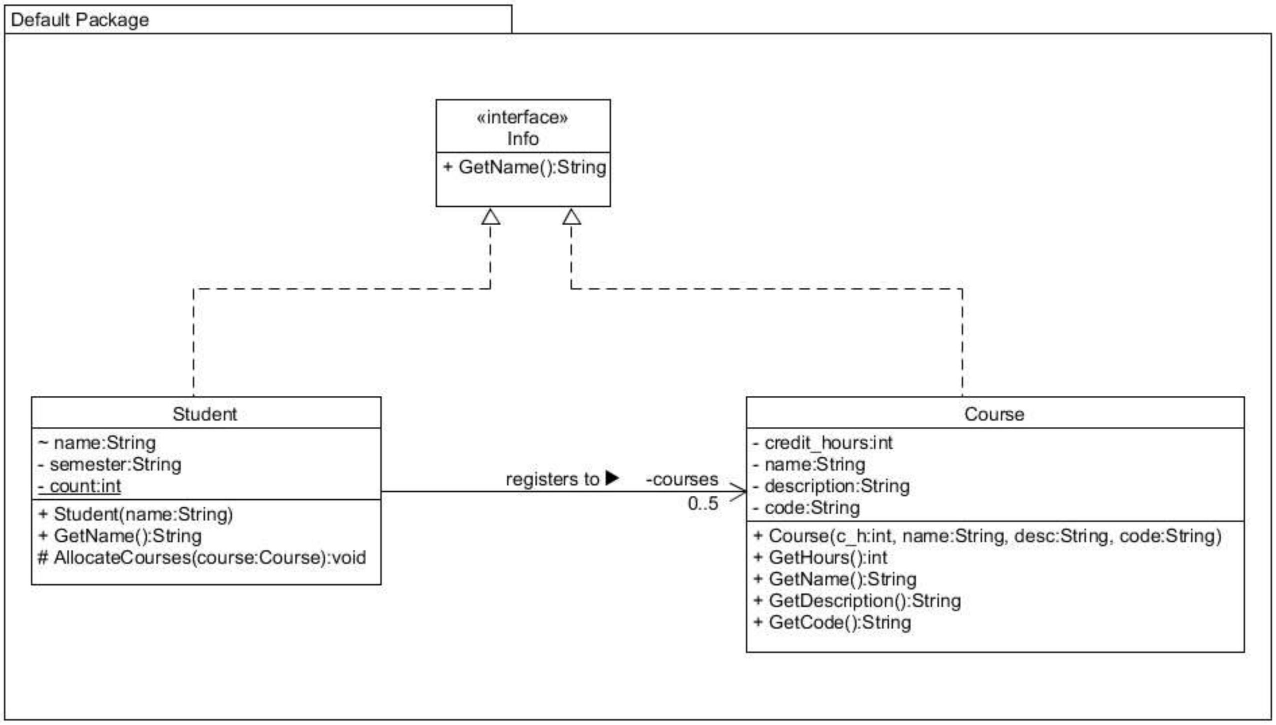
Comments for Student:

Attachments

No Attachment(s) yet

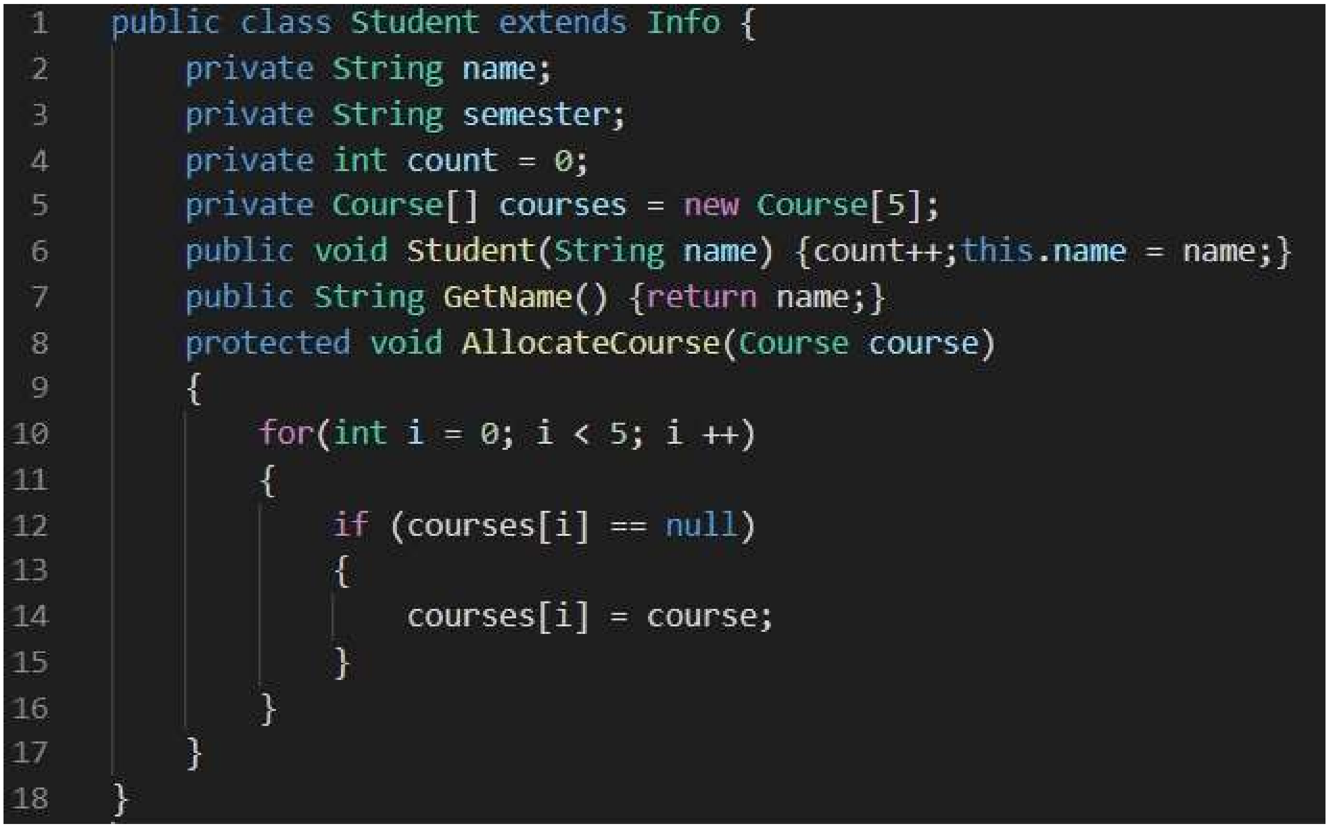
Add Attachments

- Question 5 of 5: 0.0 / 10.0 Points



Look at the class diagram above carefully. Then look at the code given below and mark the lines where you find errors. Keep in mind that if you select a wrong option, it will negate a

right option that you have selected.



```java
public class Student extends Info {
    private String name;
    private String semester;
    private int count = 0;
    private Course[] courses = new Course[5];
    public void Student(String name) {count++;this.name = name;}
    public String GetName() {return name;}
    protected void AllocateCourse(Course course)
    {
        for(int i = 0; i < 5; i ++)
        {
            if (courses[i] == null)
            {
                courses[i] = course;
            }
        }
    }
}
```

□ A. 1

□ B. 2

□ C. 3

□ D. 4

□ E. 5

□ F. 6

□ G. 7

□ H. 8

Answer Key: A, B, D, F

Comments for Student:

Attachments

No Attachment(s) yet

Add Attachments

Email REYYAN

Update Cancel

- Gateway
- Accessibility Information
- The Sakai Project
- ℹ

Sakai 19.1

✖

Wed, 22 Dec 2021 18:01:37 PKT

Server:

SAKAI-UPGRADE

Build Info:

Sakai

- Powered by Sakai

Change Profile Picture

Error removing image

Error uploading image

Upload Choose File No file chosen

Save Cancel Connections ✖ Remove

Search for people ...

View More

My Connections Pending Connections

You don't have any connnections yet. Search for people above to get started.

You have no pending connections.

←Back to My Connections

Search for people ...

${cmLoader.getString("connection_manager_no_results")}

Done