Encapsulation

Q 1: Define Difference between Information and Data .

Ans :

Differences between data and information:

- Data is used as input for the computer system. Information is the output of data.
- Data is unprocessed facts figures. Information is processed data.
- Data doesn't depend on Information. Information depends on data.
- Data is not specific. Information is specific.
- Data is a single unit. A group of data which carries news and meaning is called Information.
- Data doesn't carry a meaning. Information must carry a logical meaning.
- Data is the raw material. Information is the product.

Source **:** **http://www.differencebetween.info/difference-between-data-and-information**

Q 2 : What is Encapsulation ?

Ans : Encapsulation in java is a *process of wrapping code and data together into a single unit* .

Source *: https://www.javatpoint.com/encapsulation*

Q 3: Describe advantages of Encapsulation .

Ans :

Benefits of Encapsulation

1. Makes it easy to model real-world entities – hence easy to understand and maintain
2. Control the way data is accessed or modified
3. Makes the class easy to use for clients
4. Increase reusability
5. Aids to the flexibility of design

Source **:** **http://brevitaz.com/encapsulation-example-benefits-java/**

Advantages of Encapsulation:

- **Data Hiding:** The user will have no idea about the inner implementation of the class. It will not be visible to the user that how the class is storing values in the variables. He only knows that we are passing the values to a setter method and variables are getting initialized with that value.

- **Increased Flexibility:** We can make the variables of the class as read-only or write-only depending on our requirement. If we wish to make the variables as read-only then we have to omit the setter methods like setName(), setAge() etc. from the above program or if we wish to make the variables as write-only then we have to omit the get methods like getName(), getAge() etc. from the above program
- **Reusability:** Encapsulation also improves the re-usability and easy to change with new requirements.
- **Testing code is easy:** Encapsulated code is easy to test for unit testing.

Source : https://www.geeksforgeeks.org/encapsulation-in-java/

Q 4 : How can I set a variable read only in java ?
Ans : Setting only the Getter method on private variable and omitting the setter method .

Source **:**http://www.dummies.com/programming/java/how-to-create-a-read-only-property-in-javafx/

Q 5: What is implementation hiding ?
Ans : Interface and **implementation**. Access control is often referred to as **implementation hiding**. Wrapping data and methods within classes in combination with **implementation hiding** is often called encapsulation. The result is a data type with characteristics and behaviors.

Source :http://iimk.ac.in/gsdl/cgi-bin/library?e=d-000-00---0inftec--00-0-0--0prompt-10---4------0-1l--1-en-50---20-about---00031-001-1-0utfZz-8-00&a=d&cl=CL2&d=HASH013a1053348ed13cf7a48e82.8

Q 6 : Define a Encapsulation class example .
Ans : Java Bean Class **.**

Q 7 : Why do we use setter and getter method in Encapsulation ?
Ans : To prevent data misuse and unwanted change .

Q 8: What support Encapsulation ?
Ans : Class **.**

Q 9 : Can we limit the data of a field by using Getter and Setter method ?
Ans : yes **.**

Q 10 : What is the condition of achieving Encapsulation ?

Ans : Encapsulation in java is achieved by using private members and public methods.

- *Make members of a class as private.*
- *Define public setter and getter methods to modify and view the variables' values and access them outside the class only through getters and setters.*

Source : https://www.quora.com/How-do-we-achieve-encapsulation-in-java

Q 11: Encapsulation vs Abstraction in Java ?

Ans :

Abstraction	Encapsulation
1. Abstraction solves the problem in the design level.	1. Encapsulation solves the problem in the implementation level.
2. Abstraction is used for hiding the unwanted data and giving relevant data.	2. Encapsulation means hiding the code and data into a single unit to protect the data from outside world.
3. Abstraction lets you focus on what the object does instead of how it does it	3. Encapsulation means hiding the internal details or mechanics of how an object does something.
4. **Abstraction**- Outer layout, used in terms of design. For Example:- Outer Look of a Mobile Phone, like it has a display screen and keypad buttons to dial a number.	4. **Encapsulation**- Inner layout, used in terms of implementation. For Example:- Inner Implementation detail of a Mobile Phone, how keypad button and Display Screen are connect with each other using circuits.

Source : http://javarevisited.blogspot.com/2017/04/difference-between-abstraction-and-encapsulation-in-java-oop.html