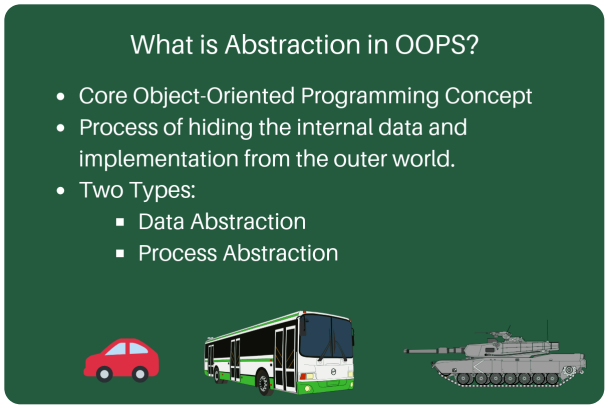
What is Abstraction in OOPS?

Pankaj - 4 Comments

Filed Under: Java

Core Java Tutorial



Abstraction is one of the core concepts of Object-Oriented Programming.
Abstraction defines a model to create an application component. The implementation of abstraction depends on the language-specific features and processes.

1. What is Abstraction?

describe things in simple terms. It's used to create a boundary between the application and the client programs.

2. Abstraction in Real Life

Abstraction is present in almost all the real life machines.

- Your car is a great example of abstraction. You can start a car by turning the key or pressing the start button. You don't need to know how the engine is getting started, what all components your car has. The car internal implementation and complex logic is completely hidden from the user.
- We can heat our food in Microwave. We press some buttons to set the timer and type of food. Finally, we get a hot and delicious meal. The microwave internal details are hidden from us. We have been given access to the functionality in a very simple manner.

3. Abstraction in OOPS

Objects are the building blocks of Object-Oriented Programming. An object

access only for required functions and properties to the other programs. This is the general procedure to implement abstraction in OOPS.

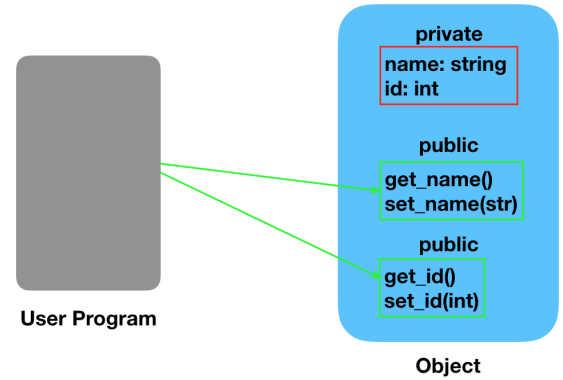
4. What are the different types of abstraction?

There are two types of abstraction.

1. Data Abstraction
2. Process Abstraction

4.1) Data Abstraction

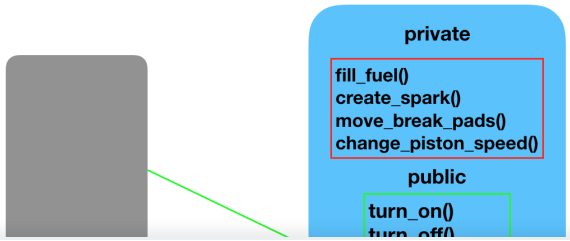
provided through some methods.



Data Abstraction

4.2) Process Abstraction

We don't need to provide details about all the functions of an object. When we hide the internal implementation of the different functions involved in a user operation, it creates process abstraction.



Process Abstraction

5. Abstraction in Java

Abstraction in Java is implemented through interfaces and abstract classes. They are used to create a base implementation or contract for the actual implementation classes.

Car.java: Base interface or abstract class

```
package com.journaldev.oops.abs

public interface Car {

        void turnOnCar();

        void turnOffCar();

        String getCarType();
}
```

ManualCar.java, AutomaticCar.java: implementation classes of Car.

```
public class ManualCar implemen

        private String carType =

        @Override
        public void turnOnCar()
                System.out.print
        }

        @Override
        public void turnOffCar(
                System.out.print
        }

        @Override
        public String getCarTyp
                return this.car
        }
}
```

```
package com.journaldev.oops.abs

public class AutomaticCar imple

        private String carType =

        @Override
        public void turnOnCar()
                System.out.print
        }
```

```
        }

        @Override
        public String getCarTyp
                return this.car
        }
}
```

User Program: Let's look at a test program where the Car functions will be used.

```
package com.journaldev.oops.abs

public class CarTest {

        public static void main
                Car car1 = new
                Car car2 = new

                car1.turnOnCar(
                car1.turnOffCar
                System.out.prin

                car2.turnOnCar(
                car2.turnOffCar
                System.out.prin

        }

}
```

details are hidden from the client program.

References: Wikipedia, Oracle Docs

What is Java Programming Language?

PREV

Java 13 Features

NEXT

Pankaj

I love Open Source technologies and writing about my

Follow Author

passion.

Comments

A. Rouabah says:
April 5, 2020 at 2:47 pm

Your definition of abstraction is definitely that of encapsulation and not of abstraction. Abstraction is a model of real-world objects limited to a specific context where we represent details related to that context with high accuracy and omit the rest!
Encapsulation is what you defined abstraction.

Thanks for the effort!

Reply

Ayorinde says:
March 26, 2020 at 3:55 pm

I am new in programming, especially OOP. Your article is good. Hope I will be able to master it.

John says: October 27, 2019 at 2:26 am

Wow. Just change relapse ‘Abstraction’ with ‘Encapsulation’ and it will be a nice article.

Reply

Micky says: September 30, 2019 at 1:19 am

Nice theme and logo sir,
Now it’s much better to study any topic

Reply

Leave a Reply

Your email address will not be published. Required fields are marked *

Comment

Post Comment

Newsletter for You

Enter your email address here...

JournalDev

JournalDev is one of the most popular websites for Java, Python, Android, and related technical articles. Our tutorials are regularly updated, error-free, and complete. Every month millions of developers like you visit JournalDev to read our tutorials.

JournalDev was founded by Pankaj Kumar in 2010 to share his experience and learnings with the whole world. He loves Open source technologies and writing on JournalDev has become his passion.

Most Popular

- Java / Java EE Tutorials
- Core Java Tutorial
- Python Tutorials
- Java Interview Questions
- Core Java Interview Questions
- Java Design Patterns
- Spring Tutorial

- GoLangDocs
- LinuxForDevices
- Mkyong
- VM-Help
- WP-Design