

OPTIONAL IN JAVA

Prepared by github.com/ali-rnp

I WHAT IS OPTIONAL ?

Java 8 has introduced a new class **Optional** in **java.util** package.

It can help in writing a neat code without using too many **null checks**.

It is a **public final class** and used to deal with **NullPointerException** in Java application.

By using Optional, we can specify alternate values to return or alternate code to run.

II HOW TO DEFINE OPTIONAL ?

of

Returns an Optional with the specified present **non-null** value.

public static **<T>** **Optional**<T> of(**T** value)

How to use ?

```
Optional<String> naser = Optional.of( value: "Naser");
```

```
Optional<String> nullVal = Optional.of( value: null);  
// throws java.lang.NullPointerException
```

ofNullable

Returns an Optional describing the specified value, if **non-null**, otherwise returns an **empty Optional**.

public static **<T>** **Optional**<T> ofNullable(**T** value)

How to use ?

```
Optional<String> Naser = Optional.ofNullable( value: "NASER");
```

```
Optional<String> nullVal = Optional.ofNullable( value: null);
```

empty

Returns an **empty Optional** instance. No value is present for this Optional.

public static **<T>** **Optional**<T> ofNullable(**T** value)

How to use ?

```
Optional<Object> empty = Optional.empty();
```

III