Enums

Monday, September 15, 2025 1:19 PM

Notes on Enum in Java

- **Definition:**
  enum (enumeration) is a special data type introduced in Java 5 to define a set of **named constants**.
- **Declaration Syntax:**

```
enum Status{  5 usages  new *

    // named constants
    Running( message: "Wait little..") , Faild( message: "Try again..") , Pending( message: "Loading..") , Success( message: "Done") ;

    // fields
    String message ;  2 usages

    // constructor
    Status(String message){  8 usages  new *
        this.message = message ;
    }

}
```

- **Key Features:**
  1. Enums are **type-safe** (you can't assign values outside the defined constants).
  2. Every constant is an **object** of the enum type.
  3. Enums are implicitly public static final.
  4. Can be used directly in **switch–case** statements.
  5. Enums can have **fields, constructors and methods** (like classes).
  6. They **cannot be extended** but can **implement interfaces**.
  7. Default toString() returns the constant's name.

- Notes :
  - Named constants are in order known as ordinal
    - ```
      // ordinals
      System.out.println(s.ordinal());
      ```
    - ```
      Pending
      2
      ```
  - Switch can be used to access enum more effectively than if-else ladder
    - Here switch knows that s is a Status enum type thus there is no extra need for Status.Running

```
// accessing with switch
switch (s){
    case Running -> {
        System.out.println("getting data...");
        break;
    }
    case Faild -> {
        System.out.println("sorry! try again.");
        break;
    }
    case Pending -> {
        System.out.println("sending request.");
    }
    default -> {
        System.out.println("Done");
    }

}
```

  - Every enum is class extended from Enum class itself
  - In result , it provide methods like values() , ordinal() defined in Enum class

-
```java
// accessing all named const
Status[] service = Status.values() ;

for (Status i : service){
    System.out.println(i + " : " + i.message);
}
```

-
```
Running : Wait little..
Faild : Try again..
Pending : Loading..
Success : Done
```