IT22030

1. When to use Interface and Abstract class

Story: "Tangail Transport"

In a city called Tangail:

- All moving things (like robots, horses) can move and stop, even if they are different. So, they use an Interface.
- All vechicles (cars, trucks, bikes) are similar. So, they share common features in an Abstract class.

## Interface Example :-

```
interface moveable {
    void move();
    void stop();
}
Class Robot implements Moveable {
Public void move() { system.out.Println("Robot moving"); }
Public void stop() { system.out. println("Robot stopped"); }
}
class Horse implements Moveable {
Public void move() { system.out.Println("Horse moving"); }
Public void move() { system. out. Println("Horse stopped"); }
}
```

## III. Abstract class Example:

```
abstract class vehicle {
  string band;
vehicle (string brand) {this.brand = brand; }
void startEngine () { system.out.println (brand + "started"); }
  abstract void drive ();
}
class car extends vehicle {
car (string brand) { super (brand); }
void drive () { system.out.println (brand + "car driving"); }
}
}
```

* Is interface slower than Abstract class?

- Yess. a little slower because interface methods are found at runtime.
- but in real program, this diffrence is very small.

## * Abstract vs Interface :

| Feature | Abstract class | Interface |
|---|---|---|
| Inheritance | single | multiple |
| methods | can have code | only method names |
| variables | can have normal variable | only constant |
| constructors | yes | No |
| performance | slightly faster | slightly slower |