

Exception Handling

Exception:- Exception is an unexpected or abnormal condition that disrupts the normal flow of the program.

Exception Handling:- It is a mechanism to handle the disruption caused by an event in the normal flow of the code/program.

Real Life Example:-



② At the top, it started to rain. Soon after the dinner, he felt severe headache.



③ But he was prepared for this situation, as he already brought required medicines. So he knows how to recover from the situation, or he knows how to handle the situation.

↓
(Event Handling)

Event Handling :- (In Real Life)

```
if (situation-1)  
{
```

```
}
```

```
then {
```

- this will be done

```
}
```

Exception Handling :-

```
try {
```

```
    Risky Code;  
}
```

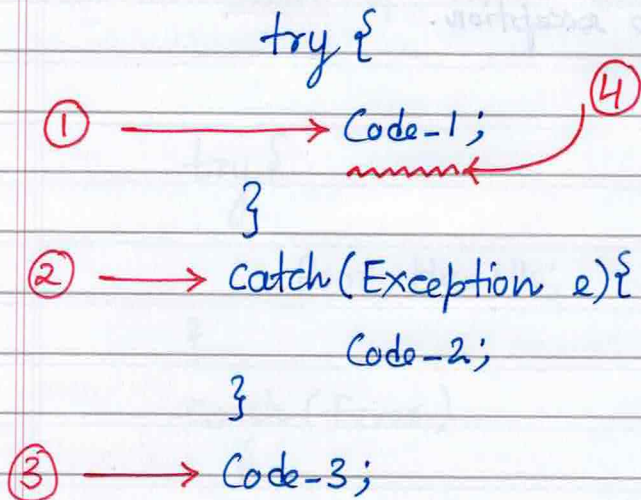
```
Catch (Exception e) {
```

Do this; // to handle or recover from the situation

```
}
```

Bullet Points :-

- * Put risky code in try block.
- * put handler code or recovery code in catch block.
- * if risky code will fail then catch block will be executed
- * if risky code will not fail then catch block will not be executed

Execution x Flow x Control(A) Code-1 (Failed)

- ① → ② → ③

(B) Code-1 (Not Failed, OK)

- ① → ③
- Catch block will not be executed. (Not required)

Note :-

- if Code-1, breaks/fails then anything written after Code-1, in try block ④, will not be executed.