

• Switch statements :

switch (expression) {

Case one :

//code block

break; → terminate the sequence

Case two :

//code block

break;

default :

//code block

}

→ default will execute when none of above does

→ if default is not at end. put break after it.

→ if break is not used then it will continue with other cases

→ duplicate cases not allowed.

eg. case one :

```
// code block
break;
```

case one :

```
// code block
break;
```

X
Not allowed.

* New Syntax:

```
switch(expression) {
```

```
  case one → // do this ;
```

```
  case two → // do this ;
```

```
  default → // do this ;
```

```
}
```

* `x.equals("word")` → here equals only checks value not reference.

`x == "word"` → here it checks reference

* Nested switch case:-

```
switch(expression) {
```

```
  case one :
```

```
    // code block
```

```
  case two :
```

```
    switch(expression) {
```

```
      case one :
```

```
        // code block
```

```
        break;
```


case two:

```
//code block  
break;
```

default:

```
//code block
```

```
}
```

```
break;
```

default:

```
//code block
```

```
}
```