Control structures

- A control structure refers to the way in which the programmer specifies the order of executing the statements
- The following approaches can be chosen depending on the problem statement:

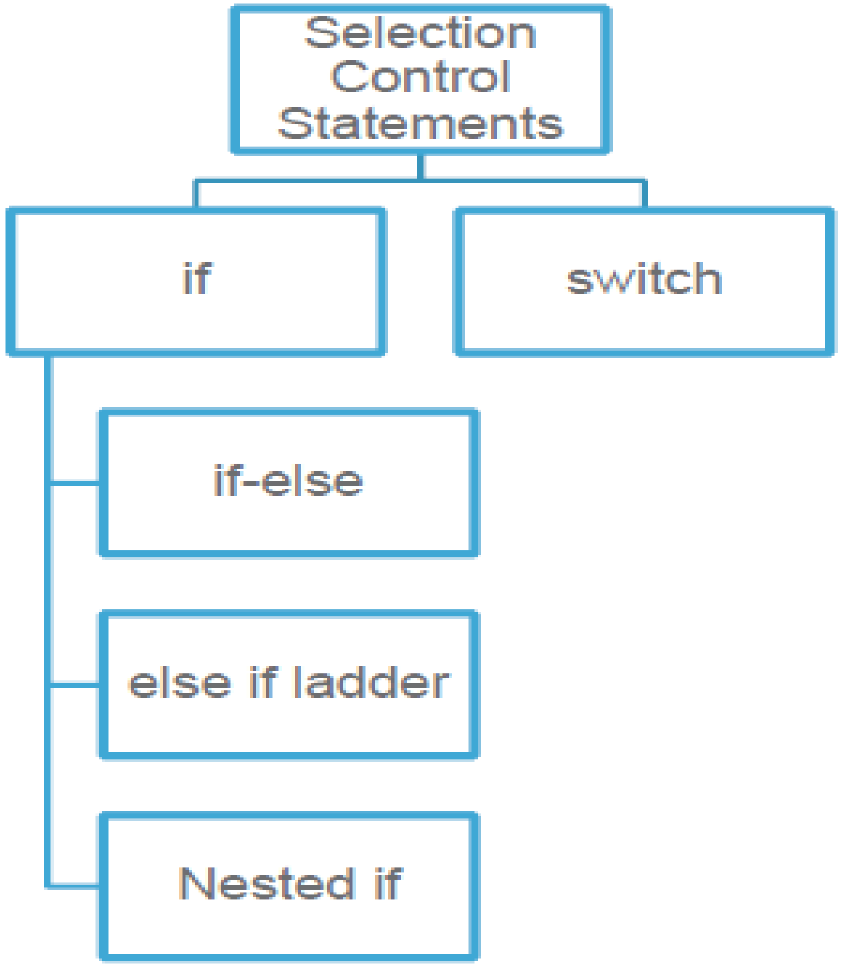
Sequential	• All the statements will be executed in the same order as it is written
Selection	• Based on some conditions, different set of statements will be executed
Iteration	• Certain statements will be executed repeatedly

Control statements

Decision making and branching

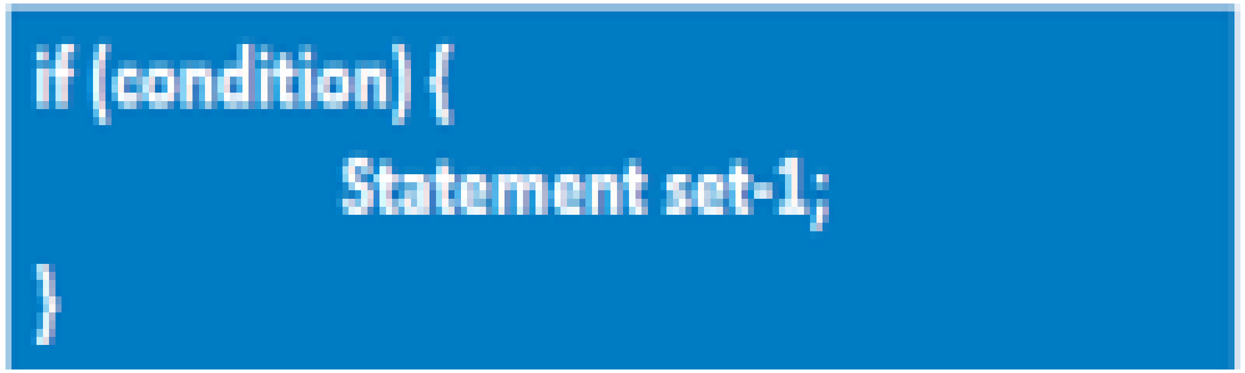
- ❑Simple if statement
- ❑Nested if   statement
- ❑If else statements
- ❑Ifelse ladder
- ❑Switch statements.

Selection Control structures

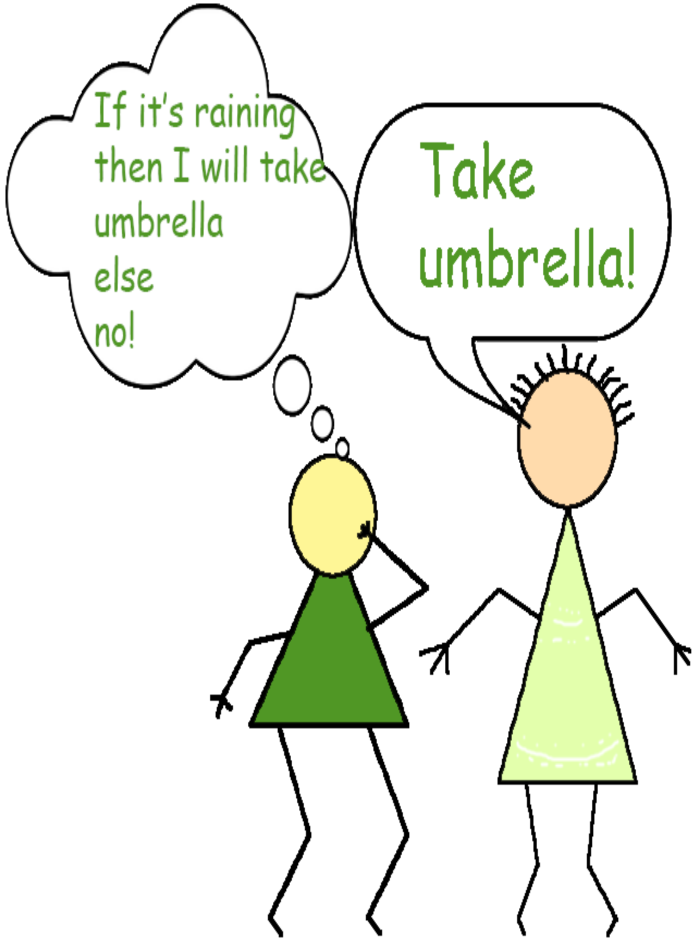


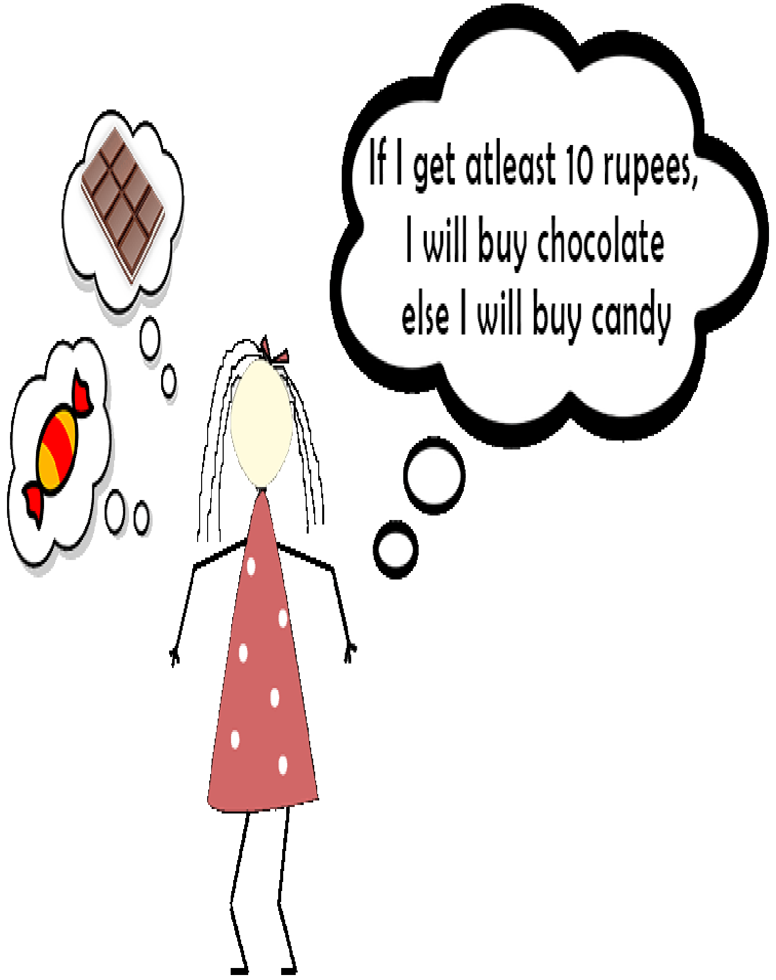
Selection Control structures statements

Simple if statement

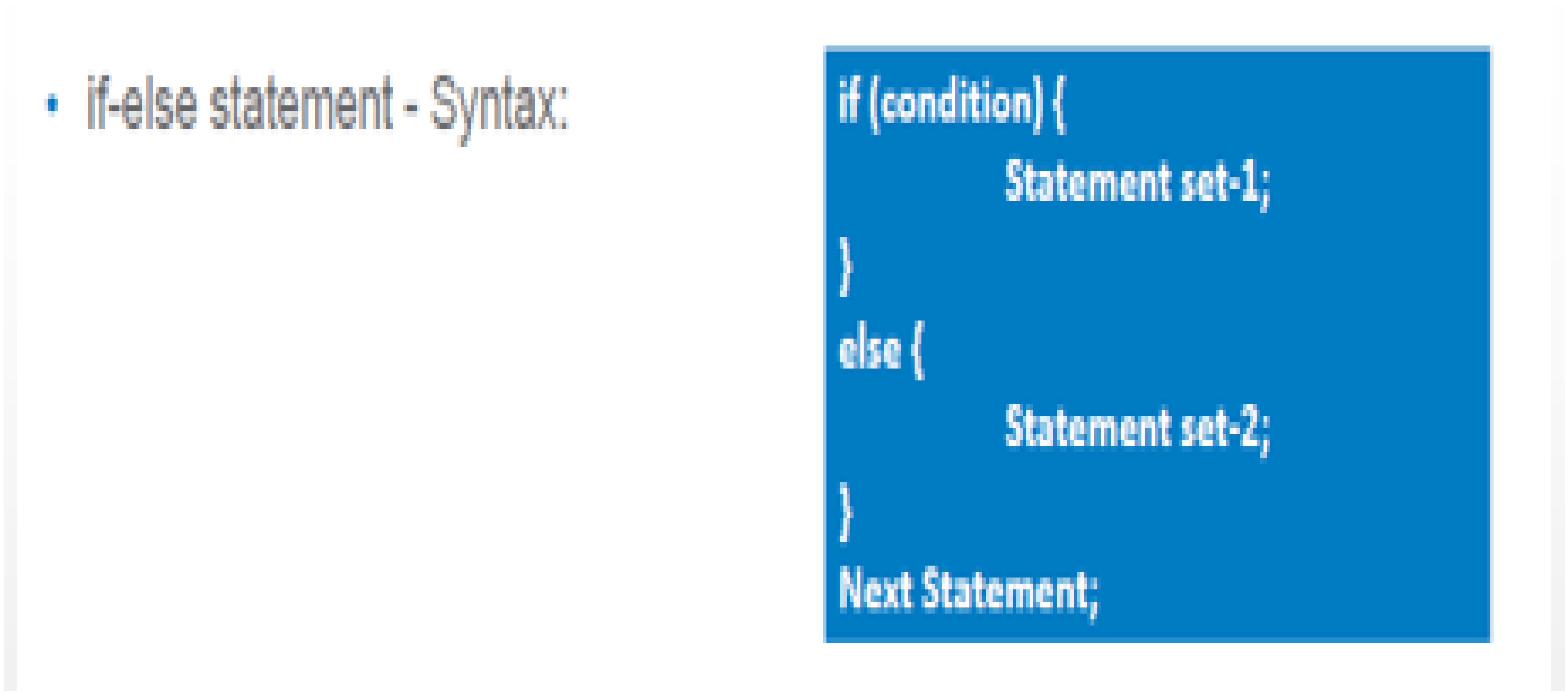


```
if (condition) {
            Statement set-1;
}
```


If it's raining then I will take umbrella else no!
Take umbrella!


If I get atleast 10 rupees, I will buy chocolate else I will buy candy

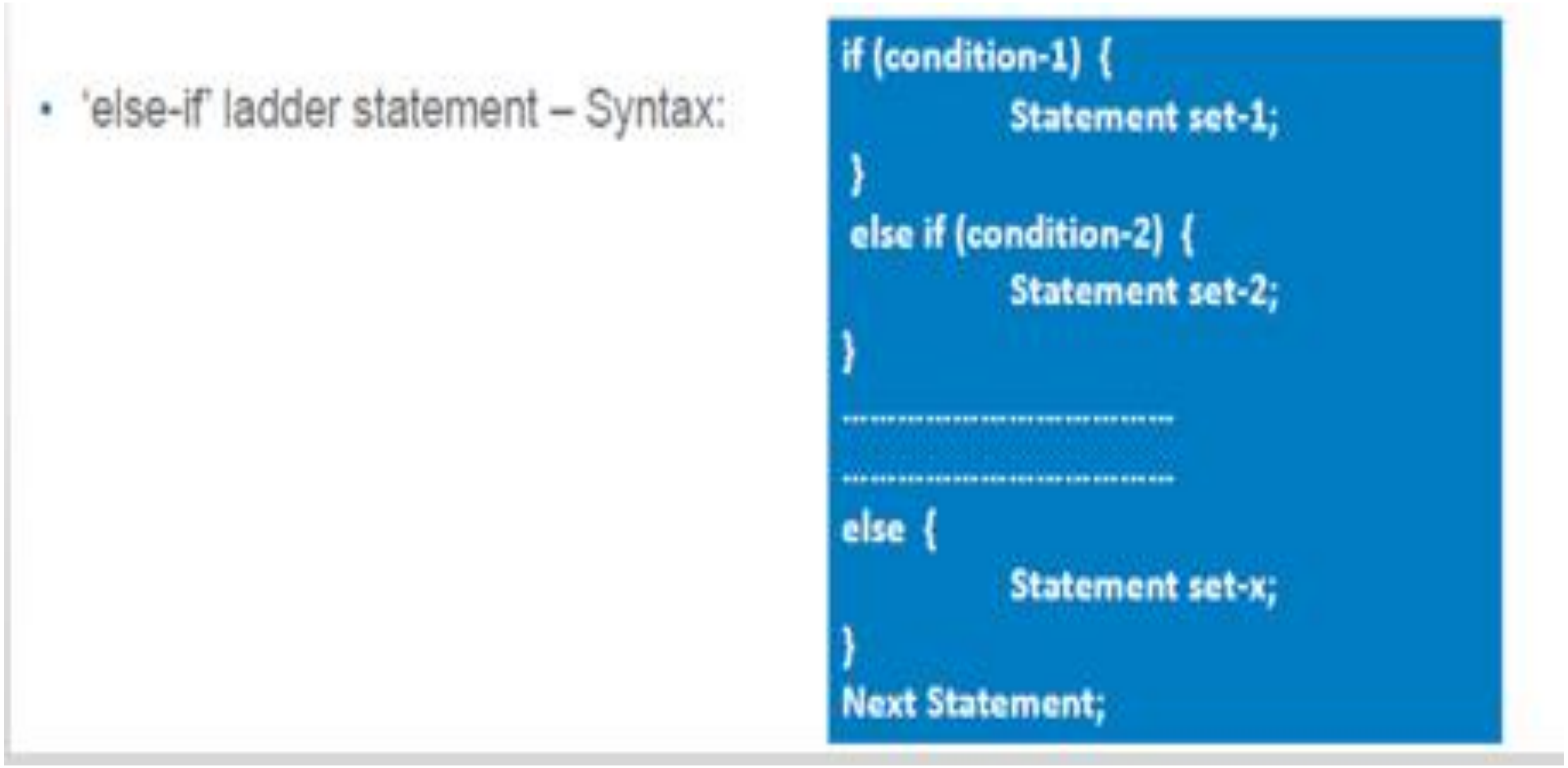
Selection Control structures statements



- if-else statement - Syntax:

```
if (condition) {
            Statement set-1;
}
else {
            Statement set-2;
}
Next Statement;
```

Selection Control structures statements



- 'else-if' ladder statement – Syntax:

```
if (condition-1) {
            Statement set-1;
 }
 else if (condition-2) {
            Statement set-2;
}
…………………………………………
…………………………………………
else {
            Statement set-x;
}
Next Statement;
```

Selection Control structures statements

- Nested if statement - Syntax:

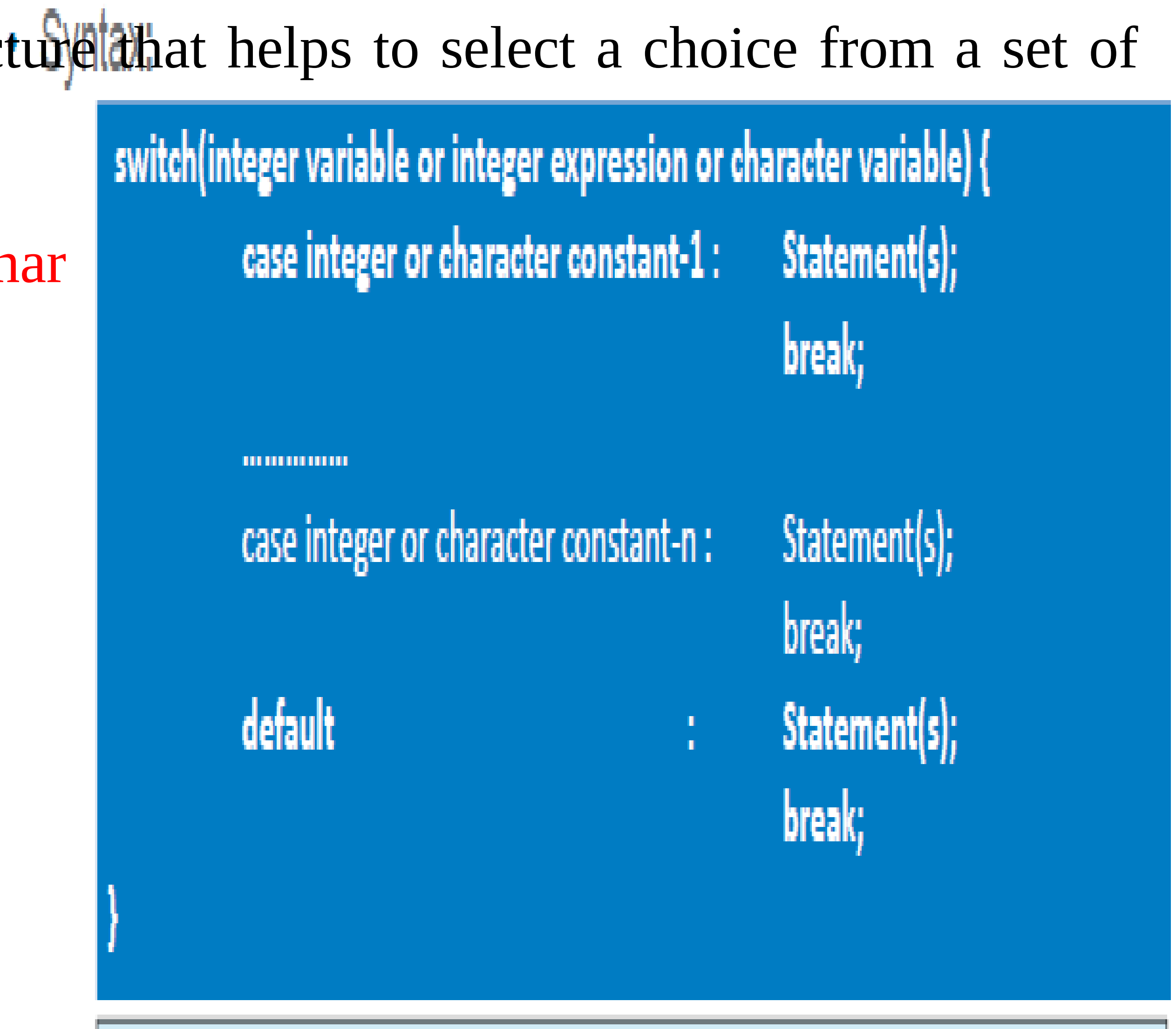
```
if (condition-1)  {
          if (condition-2) {
              .....
              if(condition-n){
                 Statement set-1;
              }
              else{
                  Statement set-2;
               }
           }
           ..................
           else{
                 Statement set-n;
            }
}
else  {
             Statement set-x;
}
Next Statement;
```

Selection Control structures – **switch statement**

Switch statement is a selection control structure that helps to select a choice from a set of available choices.

Expression must be of type byte ,short,int or char

Case must be integer or char



Syntax:

```
switch(integer variable or integer expression or character variable) {
        case integer or character constant-1 :     Statement(s);
                                                   break;
        ……………
        case integer or character constant-n :     Statement(s);
                                                   break;
        default                              :     Statement(s);
                                                   break;
}
```

Selection Control structures - switch statement

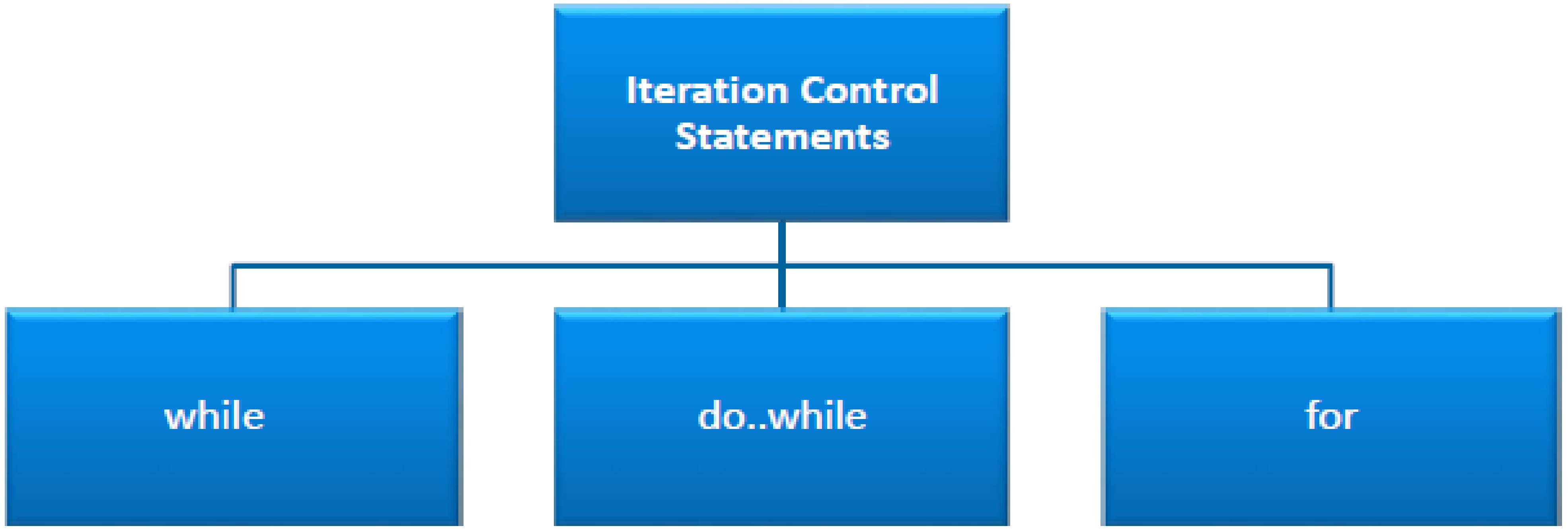
```
int number=20;
    switch(number)
{
    case 10: System.out.println("10");
                  break;
    case 20: System.out.println("20");
                 break;
    case 30: System.out.println("30");
                  break;
    default:System.out.println("Not in 10, 20 or 30");
}
```

Selection Control structures – switch statement

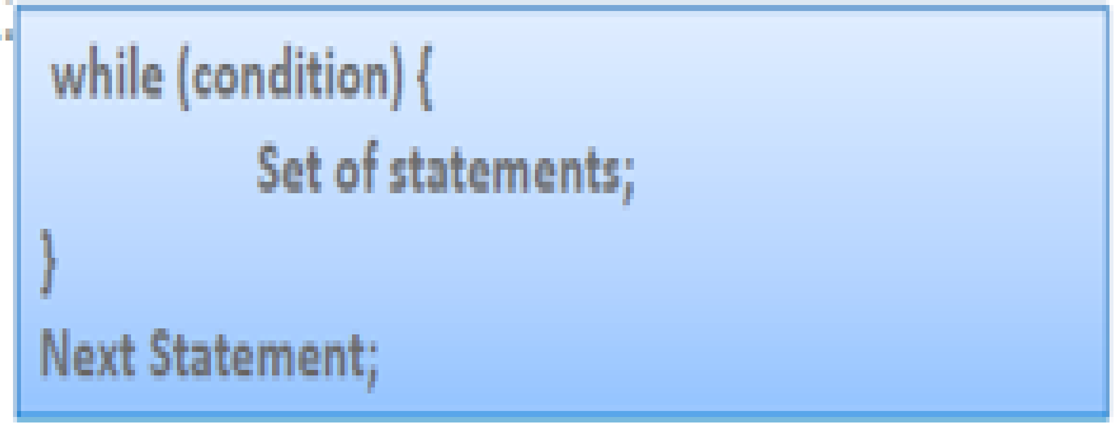
```
  int number=30;
    switch(number)
{
    case 10: System.out.println("10");
    case 20: System.out.println("20");
    case 30: System.out.println("30");
    case 40: System.out.println(“40");
    default:System.out.println("Not in 10, 20,30 or 40");

}
```

Iterational Control Structures

- Iteration (repetitive) control structures are used to repeat certain statements for a specified number of times
- The statements are executed as long as the condition is true
- These kind of control structures are also called as loop control structures

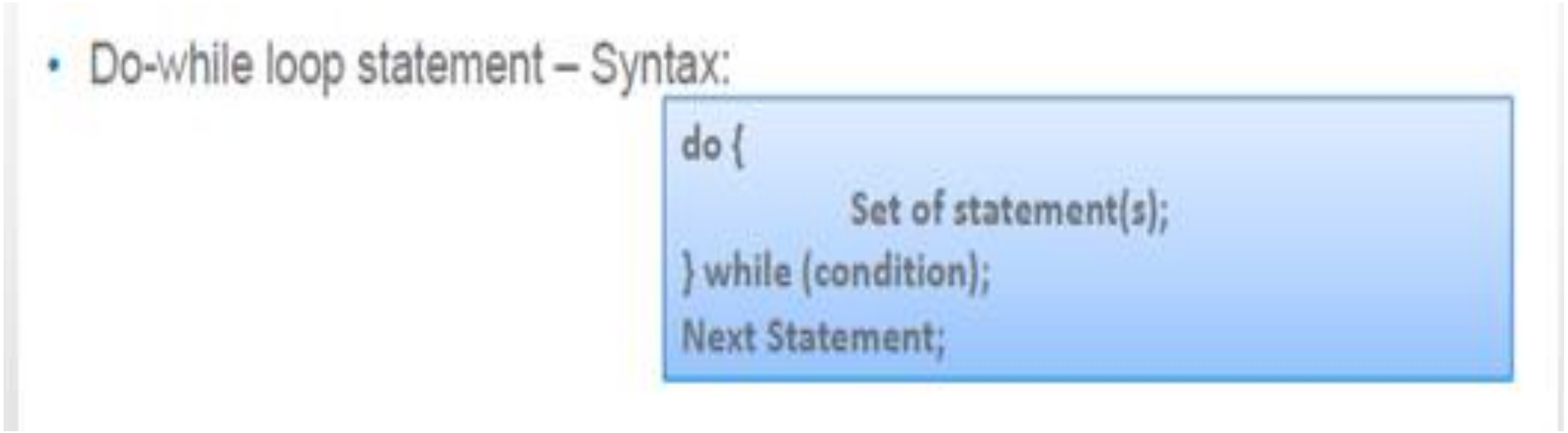


- While loop statement - Syntax:



```
while (condition) {
        Set of statements;
}
Next Statement;
```

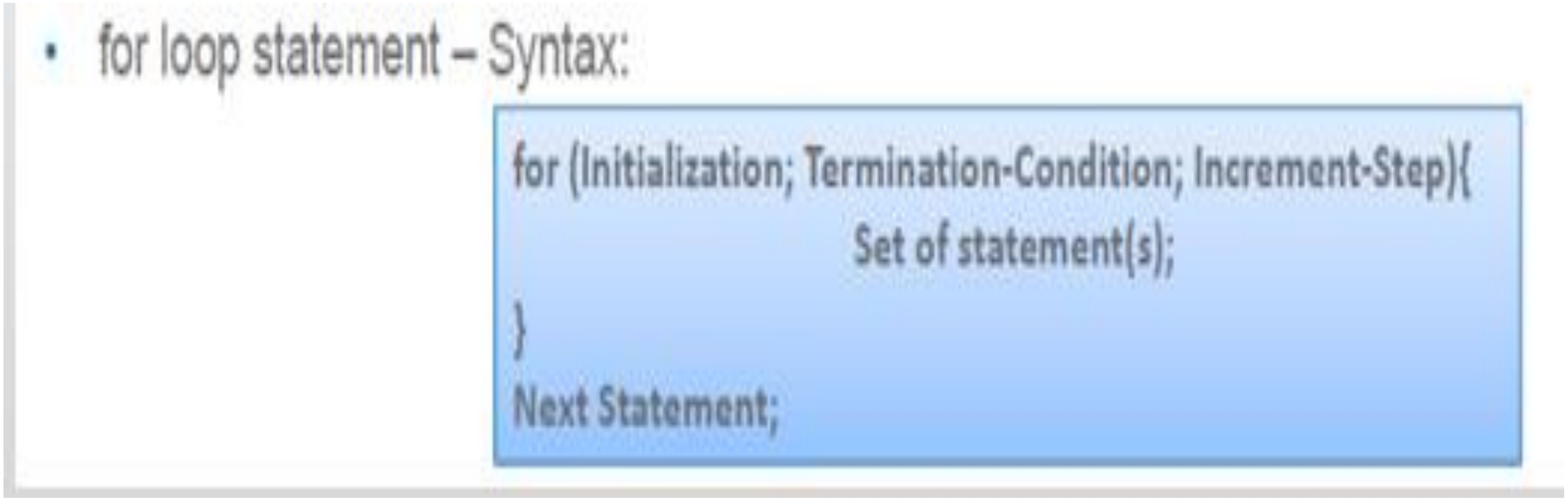
```
int count = 0;
while (count < 100)
{
  System.out.println("Welcome to Java");
  count++;
}
```



- Do-while loop statement – Syntax:

```
do {
            Set of statement(s);
} while (condition);
Next Statement;
```

```
int count = 0;
 do
{
  System.out.println("Welcome to Java!");
  count++;
} while (count < 2)
```



- for loop statement – Syntax:

```
for (Initialization; Termination-Condition; Increment-Step){
                    Set of statement(s);
}
Next Statement;
```

```
int i;
for (i = 0; i < 2; i++)
 {
  System.out.println( "Welcome to Java!");
}
```

```
for ( ; ; ) {
  // Do something
}
```

(a)

Equivalent

```
while (true) {
  // Do something
}
```

(b)

```
while (loop-continuation-condition) {
  // Loop body
}
```

(a)

Equivalent

```
for ( ; loop-continuation-condition; )
  // Loop body
}
```

(b)