LOOPS

while   do - while   for

LOOPS

while   do - while   for

```
while ( condition )
{
    /* statements
    … */

}
```

LOOPS

while

```
while ( condition )
{
    /* statements
    … */

}
```
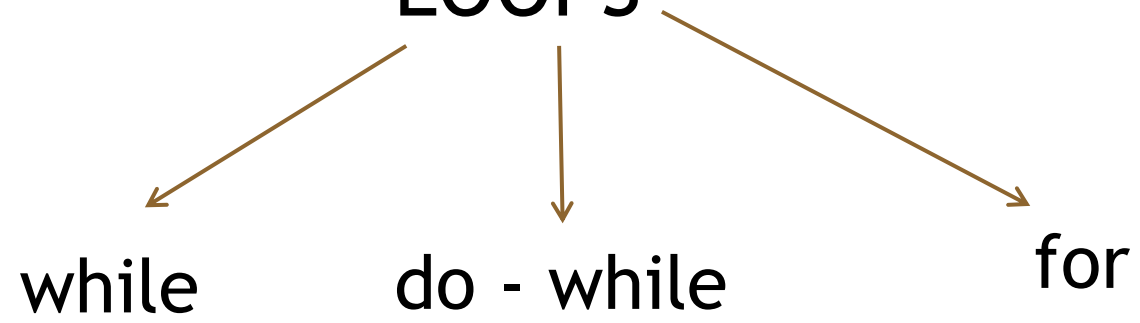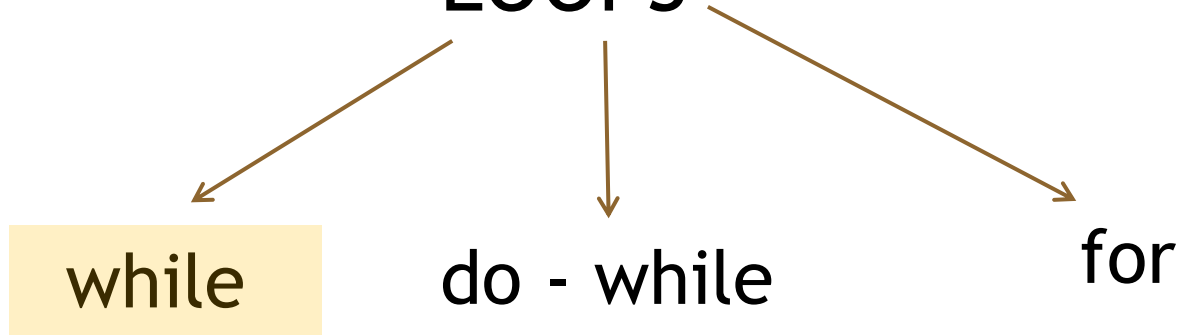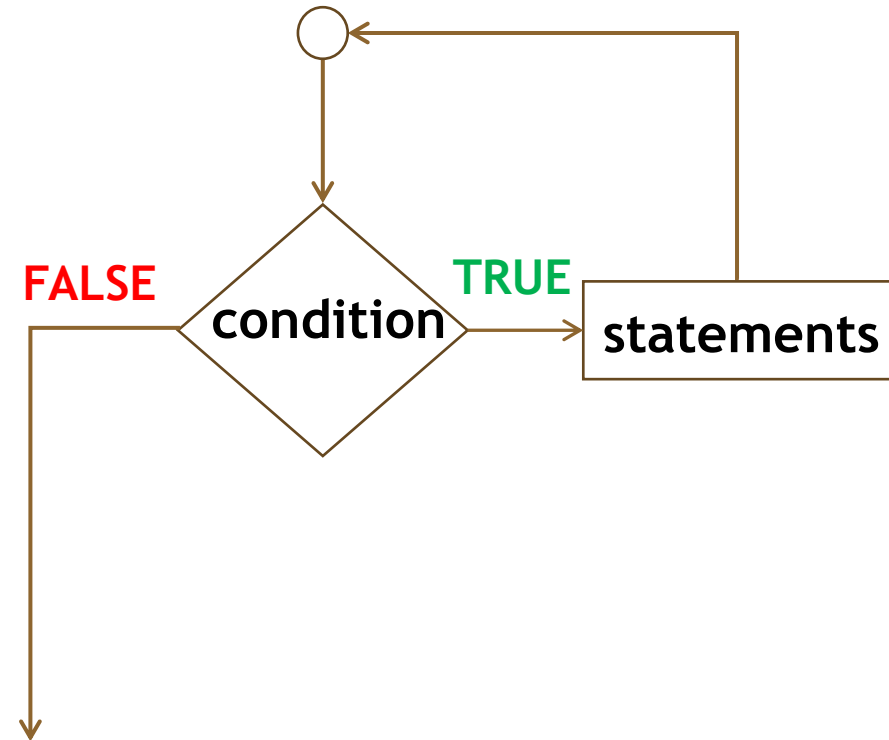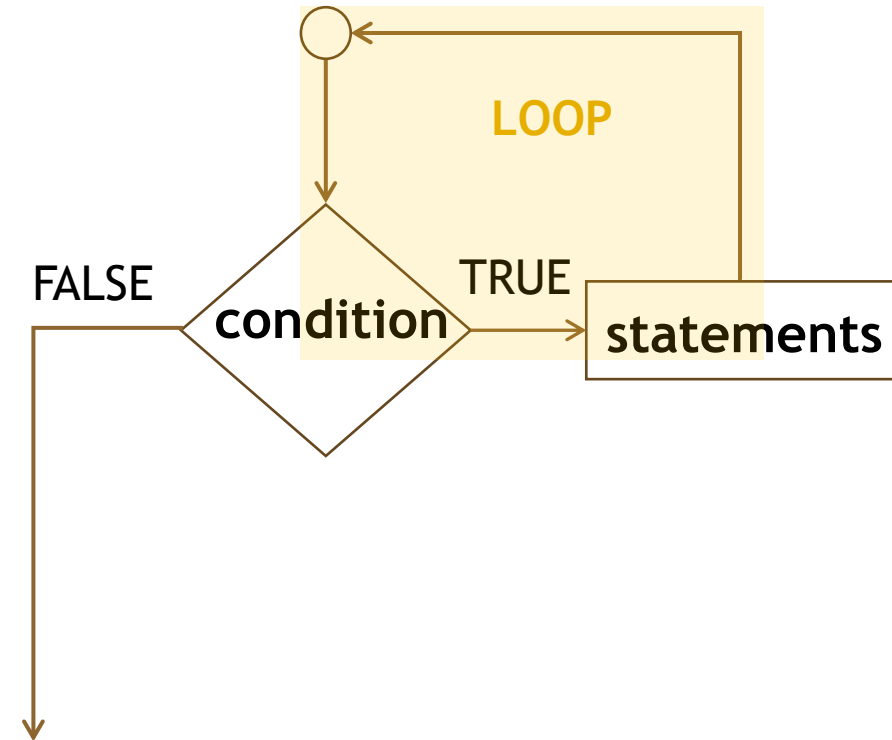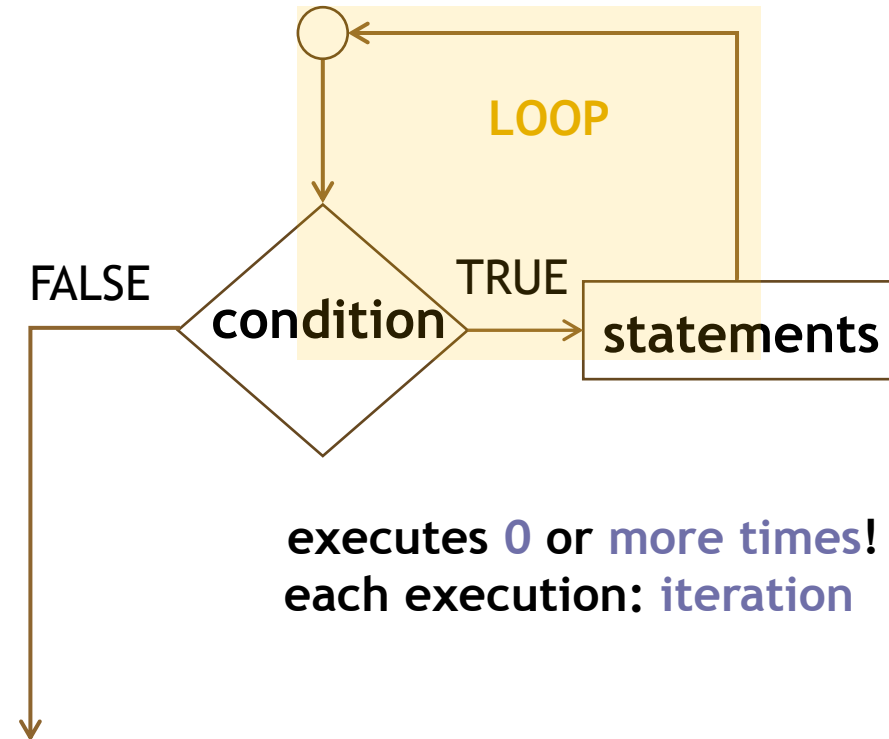
LOOPS

while

```
while ( condition )
{
    /* statements
    … */

}
```

LOOPS

while

```
while ( condition )
{
    /* statements
    … */

}
```

LOOP

FALSE   TRUE
condition → statements

executes 0 or more times!
each execution: iteration

LOOPS

```
while ( condition )    initialize the loop control variable
{
    /* statements
    … */

}
```

LOOPS

```
i = 0;
while ( i <= 10)        initialize the loop control variable
{
    /* statements
    … */

}
```

LOOPS

```
i = 0;
while ( i <= 10)
{
    /* statements
     … */
    i +=1;



}
```

update the loop control variable

IF NOT, your code might stuck in INFINITE LOOP!
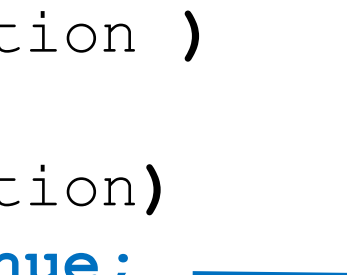
LOOPS

```
i = 0;                initialize
while ( i <= 10)      test condition
{
    /* statements
     … */
    i +=1;                update

}
```

LOOPS

while

```
while ( condition )
{
    if (condition)
        continue;


}
```
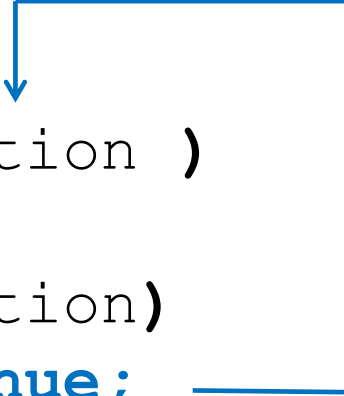
skips the rest of the loop &
continues to next iteration

LOOPS

while

```
while ( condition )
{
    if (condition)
        continue;



}
```

**skips the rest of the loop &
continues to next iteration**

```
while ( condition )
{
    if (condition)
        break;


}

// statements below
// while loop
```
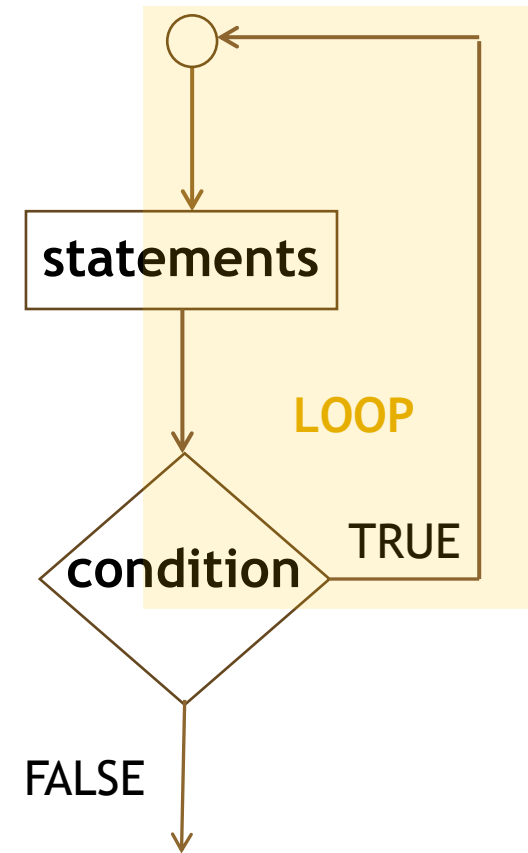
terminates the loop

LOOPS

do - while

```
do
{
    /* statements
    … */

} while ( condition )
```
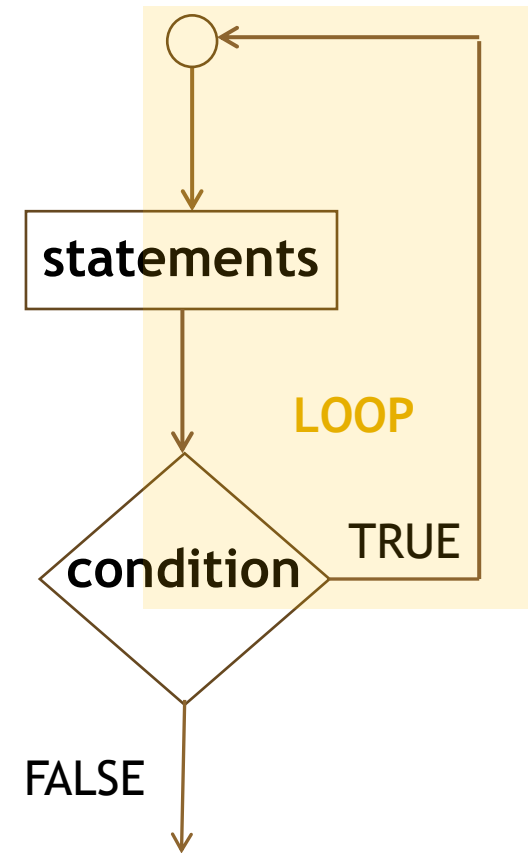
LOOPS

do - while

```
do
{
    /* statements
    … */

} while ( condition )
```

executes 1 or more times!

statements

LOOP

condition   TRUE

FALSE

LOOPS

for

```
for ( i = 1, i < n, i++)    Equal to while
{
    /* statements
    … */

}
```

LOOPS

```
for ( i = 1, i < n, i++)
{
    /* statements
    … */

}
```

initialize
test
condition
update

```
i = 0;
while ( i <= 10)
{
    /* statements
    … */
    i +=1;

}
```

initialize
test condition
update