











































































































conditions

when

- conditional expression with multiple branches
- similar to *switch* in other languages
- branches can be blocks of code
- evaluates sequentially each branch until one is true
- exhaustive - all possible cases must be covered
- can be used as a statement
































































































































```
val ects = when (grade) {  
    in (5..6) -> "E"  
    7 -> "D"  
    8 -> "C"  
    9 -> "B"  
    10 -> "A"  
    else -> "F"  
}
```

```
val result = when {  
    grade >= 5 -> "passed"  
    else -> {  
        println("reschedule exam")  
        "failed"  
    }  
}
```

Conditionals

when

```
val ects = when (grade) {  
    in (5..6) -> "E"  
    7 -> "D"  
    8 -> "C"  
    9 -> "B"  
    10 -> "A"  
    else -> "F"  
}  
  
val result = when {  
    grade >= 5 -> "passed"  
    else -> {  
        println("reschedule exam")  
        "failed"  
    }  
}
```

- conditional expression with multiple branches
- similar to *switch* in other languages
- branches can be blocks of code
- evaluates sequentially each branch until one is true
- exhaustive - all possible cases must be covered
- can be used as a statement

Loops

(do-)while

```
while(condition) {  
    /*  
    */  
}
```

```
do {  
    /*  
    */  
} while (condition)
```