

SWIFT SYNTAX cheat sheet

Adriana Belinski

Function

```
func getMood(...) -> String {  
    ...  
}
```

Return Type

```
func getMood(...) -> String {  
    ...  
}
```

Function Call

```
let mood = getMood(bpm: 90)
```

Argument Label

```
let mood = getMood(bpm: 90)
```

Argument

```
let mood = getMood(bpm: 90)
```

Property

```
var candy: Int = 7
```

Type

```
var candy: Int = 7
```

Optional Type

```
var candy: Int?
```

force unwrap property:

```
let dessert = candy!
```

Example of If-Let Else Statement

```
if let item = list.first {  
    print(item)  
} else { print("It's empty!") }
```

Ternary Operator

(condition ? expression1 : expression2)

```
if true  
if false
```

Nil-Coalescing Operator

```
let unwrappedCandy = candyName ?? "sweet"  
if let candyName != nil  
else
```

Equality Operator

```
treat1 == treat2  
treat1 != treat2  
returns true or false
```

Var vs Let

var is a variable, and it's mutable meaning its value can change.
used for flexibility

let is a constant meaning its value cannot change.
used for clarity and stability



Example of Array

```
let chocFlavors = [  
    "cherry",  
    "caramel",  
    "mint"  
]
```

accessing a property using an index-based method:
let redFlavor = chocFlavors[0]
accessing a property using a built-in array method:
let firstFlavor = chocFlavors.first

Example of Dictionary

```
let crayonBox = [  
    "red": 4,  
    "blue": 2,  
    "yellow": 3  
]
```

accessing a property with key:
let num = crayonBox["red"] // num = 4

Example of Tuple

```
let player = (  
    name: "Gigi",  
    level: 3,  
    score: 10  
)
```

tuple without optional labels:
let player = ("Gigi", 3, 10)

Example of Enum

```
enum Flavors {  
    case cherry  
    case banana  
    case grape  
}
```

creating a variable with an enum case:

```
var favoriteFlavor: Flavors = .banana  
assigns "banana" case to "favoriteFlavor"
```