

Модуль 1 - урок 2:

Операторы

Оператор присваивания

Используйте оператор `=` для присваивания значений

```
var favoritePerson = "Люк"
```

Используйте оператор `=` для изменения или переприсваивания значений

```
var shoeSize = 8  
shoeSize = 9
```

Арифметические операторы

Можно использовать операторы `+`, `-`, `*`, `/` и `%` для вычислений

```
var opponentScore = 3 * 8  
var myScore = 100 / 4
```

Можно использовать значения других переменных

```
var totalScore = opponentScore + myScore
```

Или значение той переменной, которую вы изменяете

```
myScore = myScore + 3
```

Базовая арифметика

Рациональные числа по умолчанию используют тип Double

```
let totalDistance = 3.9
var distanceTravelled = 1.2
var remainingDistance = totalDistance - distanceTravelled
print(remainingDistance)
```

Базовая арифметика

```
let x = 51  
let y = 4  
let z = x / y  
print(z)
```

Базовая арифметика

Использование значений Double

```
let x: Double = 51  
let y: Double = 4  
let z = x / y  
print(z)
```

12.75

Составной оператор присваивания

```
var myScore = 10  
myScore = myScore + 3
```

```
myScore += 3  
myScore -= 5  
myScore *= 2  
myScore /= 2  
myScore %= 7
```

Порядок операций

1. ()

2. * / %

3. + -

```
var x = 2
var y = 3
var z = 5
print(x + y * z)
print((x + y) * z)
```

17

25

Приведение числовых типов

```
let x = 3  
let y = 0.1415927  
let pi = x + y
```

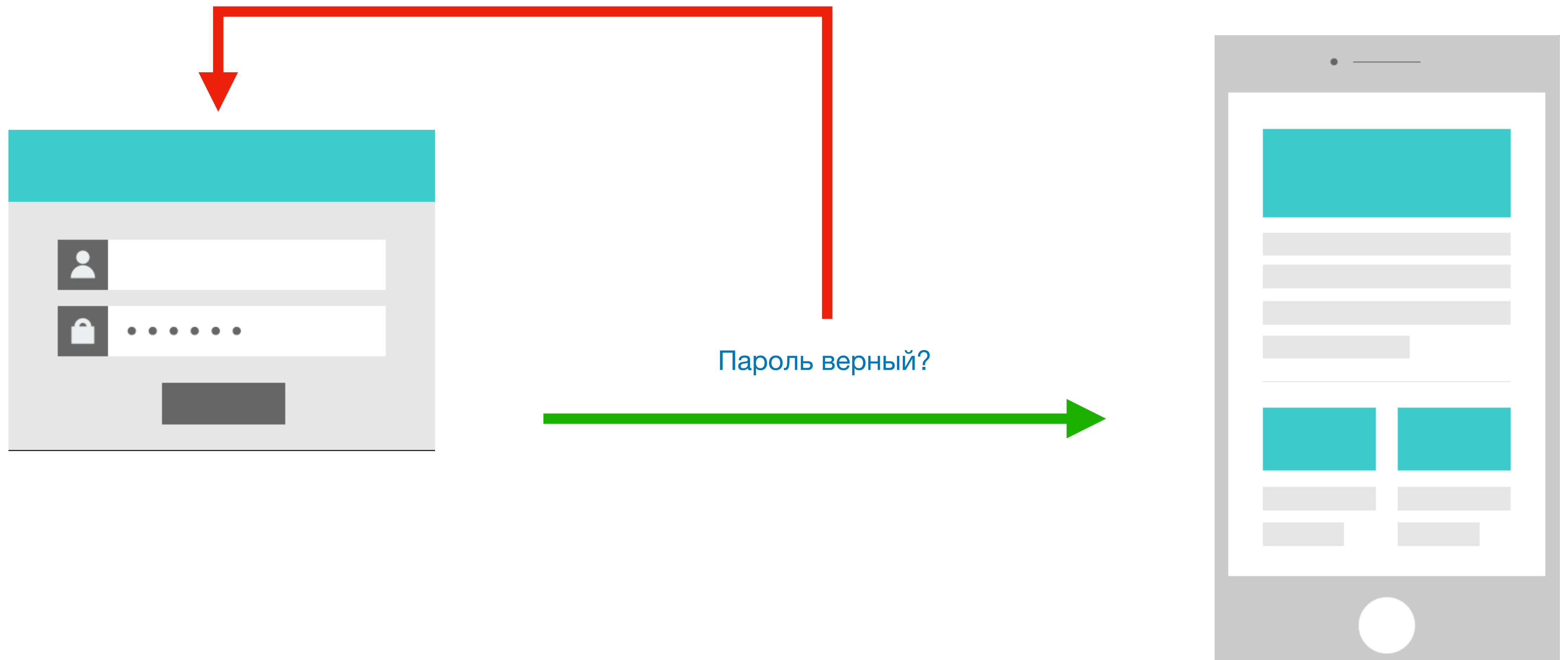


Binary operator '+' cannot be applied to operands of type 'Int' and 'Double'


Модуль 1 - урок 4:

Управляющая логика программы

Условный переход



Логические операторы

Оператор	Описание
	Значения должны быть равны
!=	Значения должны быть не равны
>	Значение слева должно быть больше значения справа
>=	Значение слева должно быть больше, либо равно, значению справа
<	Значение слева должно быть меньше значения справа
<=	Значение слева должно быть меньше, либо равно, значению справа
&&	Логическое И — оба условных выражения слева и справа должны быть истинны
	Логическое ИЛИ — хотя бы одно из двух условных выражений слева или справа должно быть истинно
!	Логическое НЕ — меняет значение следующего за оператором логического выражения на противоположное

Условная инструкция if

```
if условие {  
    код  
}
```

```
let temperature = 100  
if temperature >= 100 {  
    print("Вода кипит.")  
}
```

Вода кипит

Условная инструкция if - else

```
if условие {  
    КОД  
} else {  
    КОД  
}
```

```
let temperature = 100  
if temperature >= 100 {  
    print("Вода кипит.")  
} else {  
    print("Вода не кипит.")  
}
```

Булевы значения

```
let number = 1000  
let isSmallNumber = number < 10    // маленькое ли число
```

```
let speedLimit = 100    // ограничение скорости  
let currentSpeed = 115  // текущая скорость  
let isSpeeding = currentSpeed > speedLimit    // превышена ли скорость
```

Логические операторы

НЕ

```
var isSnowing = false    // идёт ли снег
if !isSnowing {
    print("Снегопада нет.")
}
```

Снегопада нет.

Логические операторы

И

```
let temperature = 20
if temperature >= 18 && temperature <= 23 {
  print("Температура – в самый раз.")
} else if temperature < 18 {
  print("Слишком холодно.")
} else {
  print("Слишком жарко.")
}
```

Температура – в самый раз.

Логические операторы

ИЛИ

```
var isPluggedIn = false    // подключен ли в розетку
var hasBatteryPower = true // аккумулятор заряжен
if isPluggedIn || hasBatteryPower {
    print("Ноутбук можно использовать.")
} else {
    print("😱")
}
```

Ноутбук можно использовать.

Инструкция выбора switch

```
switch значение {  
  case n:  
    код  
  case n:  
    код  
  case n:  
    код  
  default:  
    код  
}
```

```
let numberOfWheels = 2    // количество колёс
switch numberOfWheels {
case 1:
    print("Моноцикл")
case 2:
    print("Велосипед")
case 3:
    print("Трицикл")
case 4:
    print("Квадроцикл")
default:
    print("Слишком много колёс!")
}
```

Велосипед

Инструкция выбора switch

Множественные условия

```
let character = "з"    // буква

switch character {
case "а", "е", "ё", "и", "о", "у", "ы", "э", "ю", "я":
    print("Это гласная буква.")
case "б", "в", "г", "д", "ж", "з", "й", "л", "м", "н", "р":
    print("Это звонкая согласная.")
case "к", "п", "с", "т", "ф", "х", "ц", "ч", "ш", "щ":
    print("Это глухая согласная.")
default:
    print("Это не буква.")
}
```

Это звонкая согласная.

Инструкция выбора switch

Диапазоны

```
let distance = 25    // расстояние

switch distance {
case 0...9:
    print("Пункт назначения рядом.")
case 10.. $<100$ :
    print("До пункта назначения осталось немного.")
case ...1000:        // работает, начиная с Swift 4
    print("До пункта назначения ещё далеко.")
default:
    print("Вы уверены, что хотите отправиться в путешествие?")
}
```

Какой принт сработает ?

Инструкция выбора switch

Диапазоны

```
let distance = 25    // расстояние

switch distance {
case 0...9:
    print("Пункт назначения рядом.")
case 10.. $<100$ :
    print("До пункта назначения осталось немного.")
case ...1000:        // работает, начиная с Swift 4
    print("До пункта назначения ещё далеко.")
default:
    print("Вы уверены, что хотите отправиться в путешествие?")
}
```

До пункта назначения осталось немного.

Инструкция выбора switch

where и fallthrough

```
let number = -2    // число

switch number {
case let n where n % 2 == 0:
    print("Чётное")
    fallthrough    // провалится дальше без проверки
case 0...Int.max:
    print("Натуральное, включая нуль")
default:
    print("Отрицательное")
}
```

Чётное

Натуральное, включая нуль

Тернарный оператор

```
let largest: Int    // наибольшее
let a = 15
let b = 4

if a > b {
    largest = a
} else {
    largest = b
}
print(largest)
```

Тернарный оператор

? :

переменная = условие ? значение_при_истине : значение_при_лжи

```
let a = 15
let b = 4
let largest = a > b ? a : b

print(largest)
```