Основы Swift

Generic Constraints

Использование constraints

```
struct StackOfGenerics<ContainedType:
Equatable> { ... }

extension StackOfGenerics where
ContainedType: Equatable {
    func isItemOnTop(_ item: ContainedType)
    -> Bool {
        return items.last == item
    }
}
```

Дополнительные условия

```
extension StackOfGenerics where ContainedType:
Equatable {
    func isCollectionEqualToStack<CollectionType:
    Collection>(_ newColection: CollectionType) -> Bool
        where CollectionType.Iterator.Element ==
        ContainedType {
            guard items.count ==
            newColection.count as! Int else { return false }
            for (index, item) in newColection.enumerated() {
                if items[index] != item {
                    return false
                }
            }
            return true
    }
}
```

Использование в функции

```
func log<PlaceHolderType: UIView>(_ view:
                                  PlaceHolderType) { ... }

func log(_ view: UILabel) { ... }
```