

**iOS : Swift &  
Xcode Mastery**

**Virtual  
Internship  
Experience**



**mandiri**

# Disclaimer

“Dokumen ini memiliki hak cipta. Barang siapa yang menyebarluaskan atau menduplikasi tanpa izin dari instansi terkait dapat diproses sesuai dengan ketentuan hukum yang berlaku.”

# Outline

- 1 **Swift**
- 2 **Xcode**
- 3 **Case Study**

- Swift adalah bahasa pemrograman yang diciptakan oleh Chris Lattner di bawah naungan perusahaan Apple Inc.
- Bahasa pemrograman yang digunakan untuk membuat aplikasi di Apple *platform* ada dua, yaitu Objective-C dan Swift.

# Sintaks Swift

- Sintaks deklarasi variabel dengan Swift :

```
var variableName = <initial value>
```

Contoh :

```
var namaBuah = "Pisang"  
var jumlah = 10
```

- Sintaks pembuatan function dengan Swift :

```
func funcname(Parameters) ->  
returntype {  
    Statement1  
    Statement N  
    return parameters  
}
```

Contoh :

```
func student(name: String) -> String  
{  
    return name  
}
```

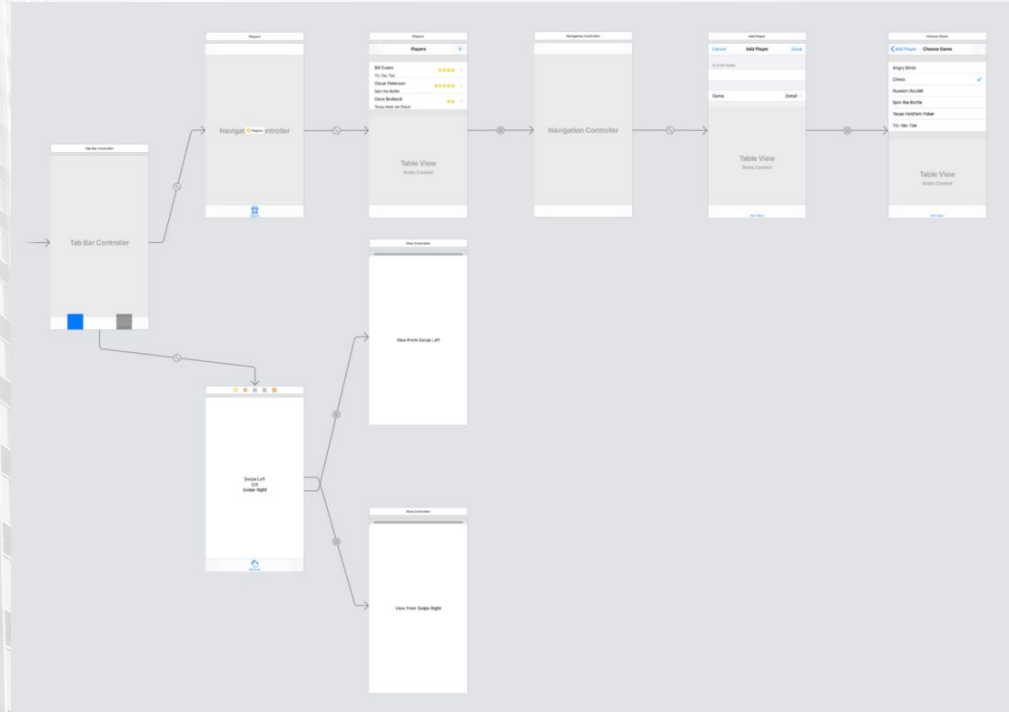


- Di lingkungan Apple, belajar Swift bisa menggunakan komputer Apple seperti MacBook, iMac, dan sebagainya dengan sistem operasi macOS terbaru dan menggunakan aplikasi Xcode.
- Xcode adalah aplikasi IDE (*Integrated Development Environment*) buatan Apple untuk menulis program dan membuat aplikasi di komputer Apple.
- Xcode mendukung beberapa bahasa pemrograman seperti C, C++, Objective C dan yang terakhir mendukung bahasa Swift yang tersedia pada Xcode 6. Seperti pada umumnya IDE, maka Xcode menyediakan semua kebutuhan pemrograman seperti editor, compiler, debugger dan run-time simulator.

# Fitur pada Xcode

- **Storyboards**

**Storyboard** adalah bagian dari **interface builder** yang telah menjadi cara paling umum untuk membuat UI untuk aplikasi iOS setelah dirilis dengan iOS 5. Setiap proyek iOS baru memiliki file `Main.storyboard` secara default.



# Fitur pada Xcode

- **SwiftUI**

**SwiftUI** adalah sebuah framework yang dikembangkan oleh Apple untuk mempermudah dalam membangun tampilan aplikasi.

Pertama kali diluncurkan bersamaan dengan iOS 13, pada September 2019.

Pendekatan yang dilakukan oleh SwiftUI adalah *Declarative Programming*, yang sudah banyak digunakan oleh platform lain seperti React Native atau Flutter.

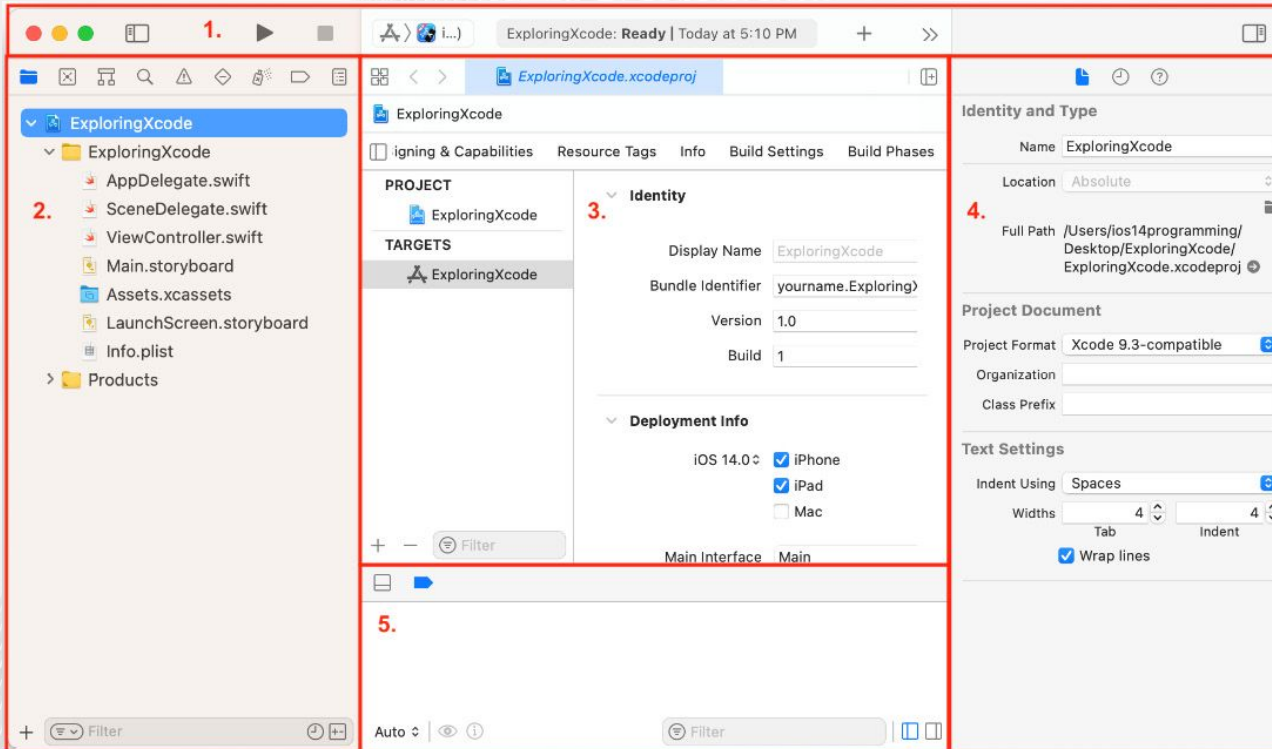
```
struct ContentView: View {  
    var body: some View {  
  
        VStack(spacing: 40) {  
  
            VStack(alignment: .leading) {  
                Text("Welcome into my app")  
                    .font(.title)  
  
                Text("Start by tapping this button")  
                    .font(.body)  
            }  
  
            Button("Start browsing") {  
                // TODO - onclick  
            }  
                .foregroundColor(.white)  
                .padding()  
                .background(Color(.blue))  
                .cornerRadius(10)  
        }  
        .padding()  
        .border(Color(.blue), width: 1)  
    }  
}
```

Welcome into my app  
Start by tapping this button

Start browsing



# Xcode User Interface



1. Toolbar
2. Navigator area
3. Editor area
4. Inspector area
5. Debug area

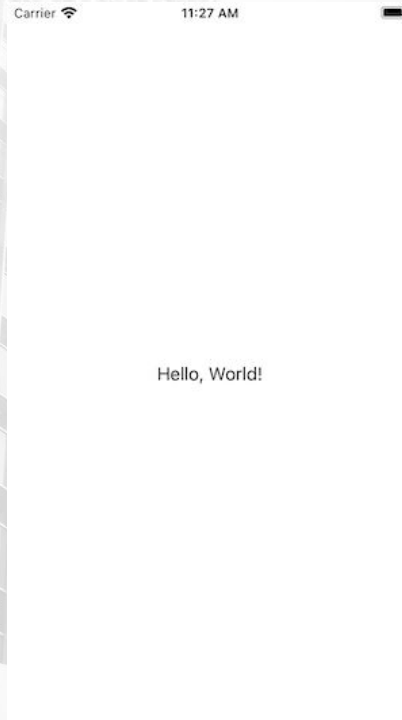
```
- (void)viewDidAppear:(BOOL)animated {  
    [self.tableView reloadData];  
    [super viewDidAppear:animated];  
}
```

## Case Study

# Membuat Aplikasi Hello World dengan SwiftUI



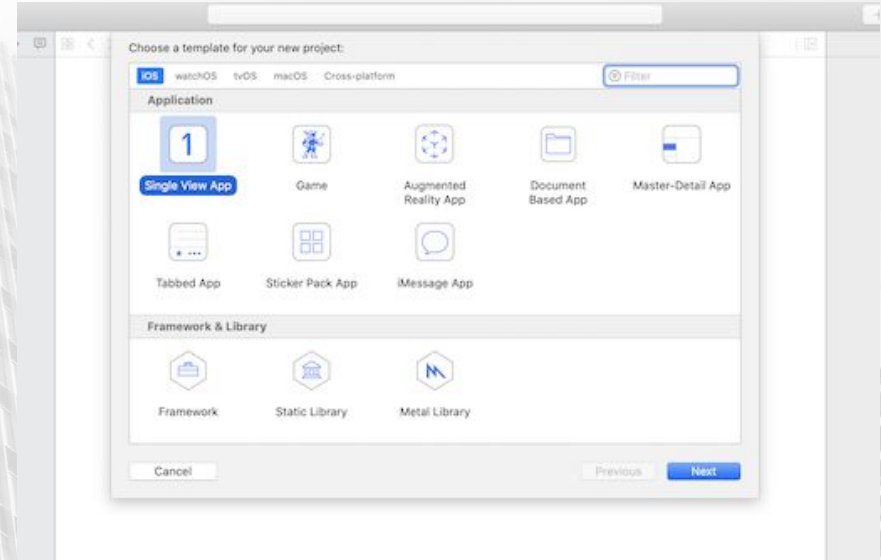
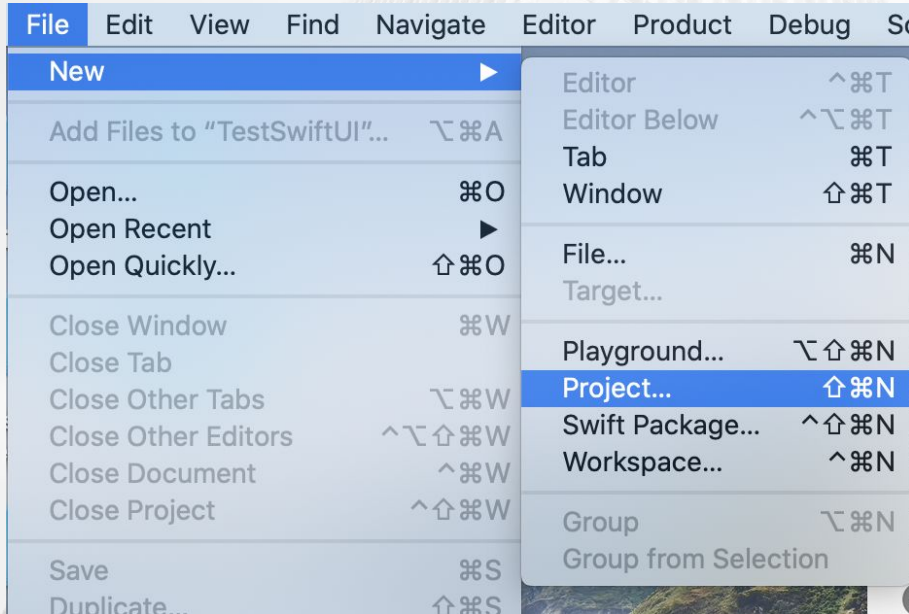
Preview aplikasi yang akan dibuat :



# Membuat Aplikasi Hello World dengan SwiftUI



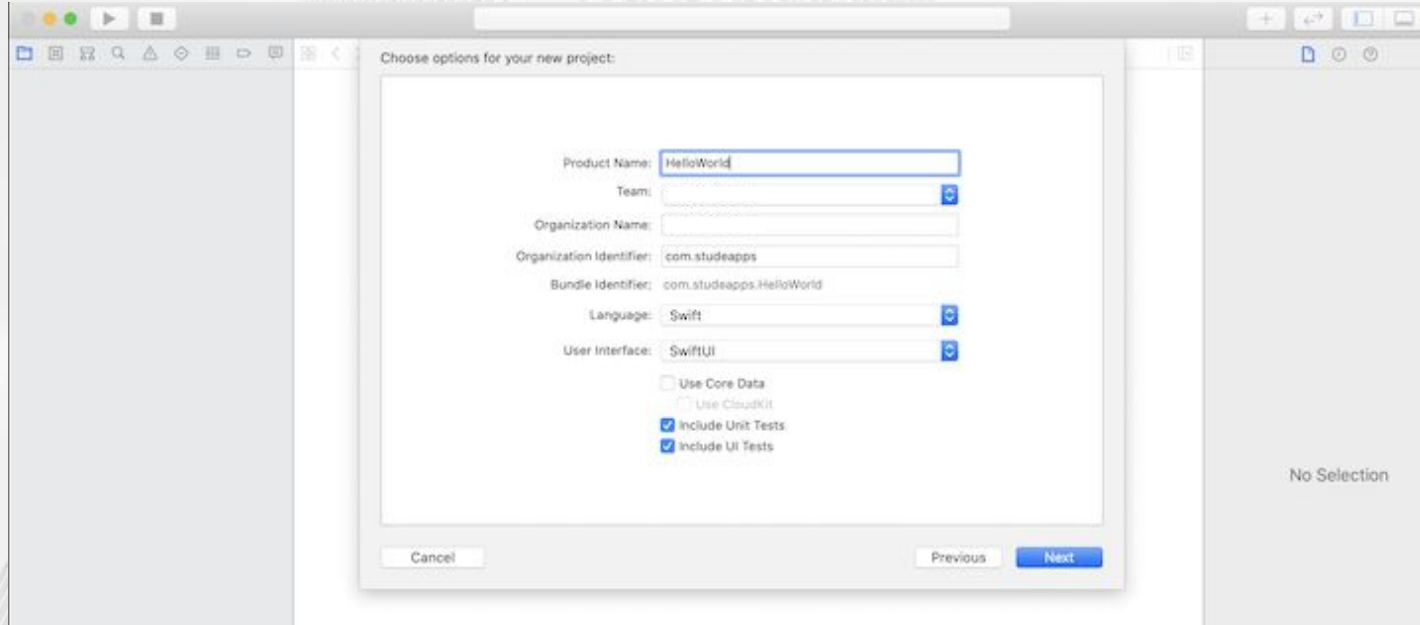
Buka Xcode kemudian pilih File -> New -> Project. Lalu pilih Single View App



# Membuat Aplikasi Hello World dengan SwiftUI



Lalu beri nama proyek, pastikan bahasanya diset sebagai Swift dan User Interface dengan SwiftUI.



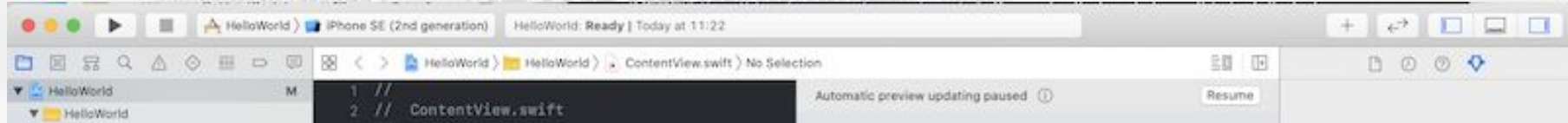


# Membuat Aplikasi Hello World dengan SwiftUI



Kita dapat melihat preview saat ini di Xcode dengan memintanya untuk memperbarui Canvas.

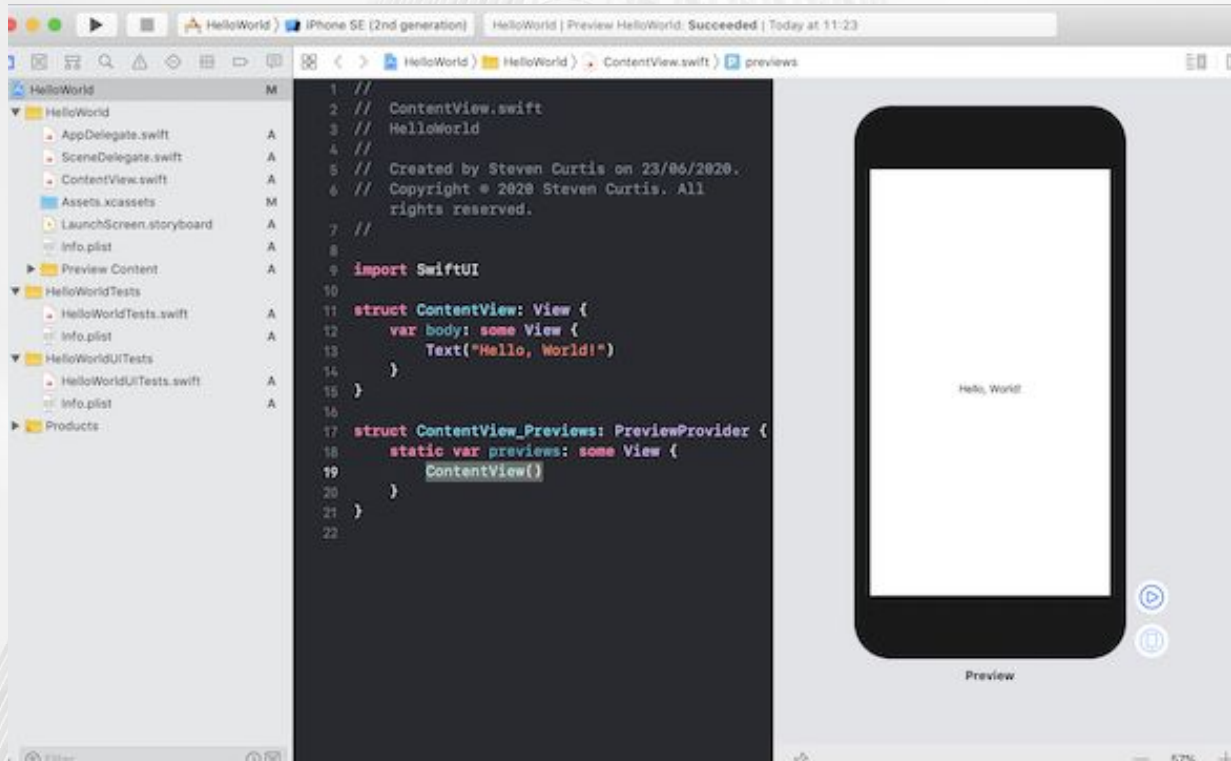
Jadi untuk melihat kanvas tekan Option-Command-Enter di keyboard. Jika Pembaruan preview otomatis di-*pause*, klik tombol *resume* di bagian atas layar untuk memulainya.



# Membuat Aplikasi Hello World dengan SwiftUI



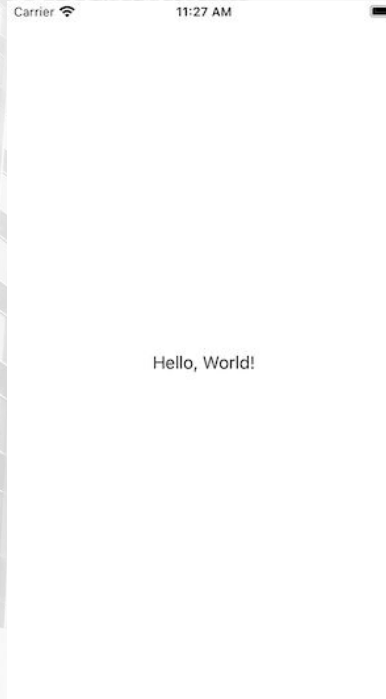
Dan setelah itu preview UI dapat terlihat.



# Membuat Aplikasi Hello World dengan SwiftUI



Untuk menjalankan kode pada simulator terdapat shortcut yaitu  dan R pada saat yang sama akan mengaktifkan simulator, membuatnya muncul dan memuat Aplikasi.





# Thank You!



mandiri