(*) Creating an Xcode Project

1. Open Xcode
2. File → New Project
3. Select IOS from window displayed
4. select single view application
5. Click next
6. Enter productName, Organization Name, identifier, language → swift, Device → universal,
   Note: Make sure that the Use Core Data checkbox is unchecked.
7. Click next and new project is created.

(*) X code workspace window

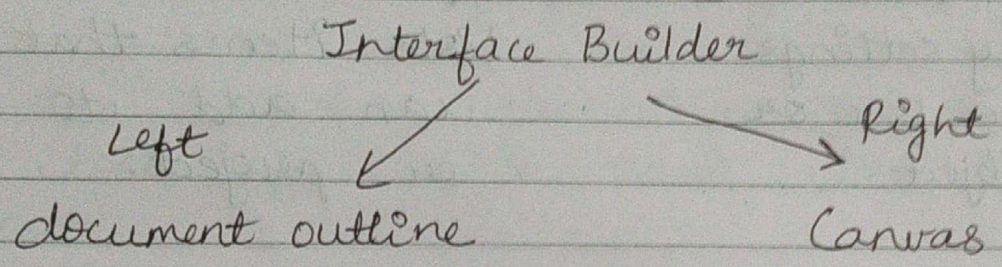
→ Lefthand side → navigation area.
↓
display different navigator tools

→ right to the navigation area is the editor area where files are opened.

⊛ Interface Builder

→ In project Navigator click on Main.storyboard. file

→ opens graphic style editor called Interface Builder.



→ lets you drag objects from a library onto the canvas to create instance and also allow establishing that connections.

→ It is a object editor that can create instances and manipulate there properties.

→ When you done editing an interface, it does not generate code that corresponds to the work you have done.

→ Storyboard. file is archive of objects to be loaded into memory when necessary.

VALENTINE
Page No.:
Date: / /

⊛ Creating View Objects.

→ Utility area → right of the editor area

(top) Inspector → display settings for a file or object

Library (bottom) → lists items that you can add to a file or project.

→ ave apde labels to add karii didha ave ena pchi apde ene configure karsu etle size, position, text, a bdha attribute directly apde change karii sakiye.

→ ave direct change karie pchi ene run karsu to eb phone mate eno view correct hse but jem bijo koi phone select karsu to eno setting misplace thai jse.

→ To a vastu nu solution che auto-Layout.

VALENTINE
Page No.:
Date: / /

26/2/2024

⊛ Auto Layout

→ IOS expects that your applications support all screen sizes and orientations.

→ To guarantee that the layout of view objects will be correct regardless of the screen size or orientation of the device running the application. The tool for this task is Auto Layout.

→ Auto Layout works by specifying position and size constraints for each view object in a scene.

→ These constraints can be relative to neighboring views or to container view.

↓
View containing another view.

Auto Layout

① Align menu → Horizonally in / Vertically in → make sure kare ke koi bhi size in screen pr Vertical ka to Horizontal center j ave.

VALENTINE
Page No.:
Date: / /

(2) Add new constraints → distance ketlu havu joie aju baju nu, width, Hight evu bdhu.

(3) Resolve auto Layout Issue → constraints ma koi mistake thai to clear karwa.

VALENTINE
Page No.:
Date: / /

✱ Application Icons

→ Select Assets.xcassets from project Navigator.
→ then select AppIcon from left side from resource list.
→ select Image from Asset Catalog.
→ drag selected image into the 2x slot of the iPhone App section.

✱ Launch Screen

→ launch image will appear while an application is loading.
→ launch image convey kare ke apdu application load thai che ane user Interface ause.
→ launch image replace thai jse after the application has launched it does not become the background image of the application.

→ project settings → select AppIcons & LaunchImage → choose main.storyboard

(*) Swift Features

1. Variables are always initialized before use.
2. Array indices are checked for Out-Of-bounds errors.
3. Integers are checked for overflow.
4. Optionals ensure that nil values are handled explicitly.
5. Memory is managed automatically
6. Error handling allows controlled recovery from unexpected failures.

VALENTINE
Page No.:
Date: / /

⊛ Optionals.

→ It is indicated by appending ? to a type name.

Eg: var anOptional Float : Float?

→ An optional let you express the possibility that a variable may not store a value at all.

→ Value can be an instance of specified type or nil.

→ Jyare apde variable na values specific rakhvi hoi jem ke ka to value ka to nil to optionals use karvanu.

→ optional variable hoi to ka to initial value ave ka to nil value ave.

→ optional ene na use kari sakiye ene unwrapping karvu pde.

optional binding

forced unwrapping.

→ forcibly unwrap karuu hoi to name pchi ! use karvanu

eg: let avgReading = (reading1! + reading2! + reading3!)/3

(reading1!, reading2!, reading3! → variables)

→ but ama issue avse if ama thi ek bhi value nil hse to error avse.

→ so ape optional binding use karvanu.
→ ema if optional nil hoi to ape ene else case thi handle karu sakiye
→ ama if-else statement ma lakhvu pde.

eg:
```
if let r1 = reading1,
   let r2 = reading2,
   let r3 = reading3 {
  let avgReading = (r1+r2+r3) /3 }
  else {
    let errorString = "Instrument was nil"
}
```

ave if nil value hse to error ni badle else statement print thse.

VALENTINE
Page No.:
Date: / /

View Hierarchy.

→ Ek main window hoi jene apde VI Window kaie.

↓

this is created when the application launches.

↓

after this other views can be added.

↳ jyare apde view add kaie to e window nu subview count thse.

↳ Main View hoi, ena subview hoi, ane ena bhi subview hoi elte ek hierarcy generate thai.

```
import UIKit

class ViewController: UIViewController {

override func viewDidLoad() {
    super.viewDidLoad()

let firstFrame = CGRect(x:160, y:240, width:100,
height:150)
let firstView = UIView(frame: firstFrame)
firstView.backgroundColor = UIColor.blue
view.addSubView(firstView)

let SecondFrame = CGRect(x:20, y:30, width:50,
height:50)
let secondView = UIView(frame: secondFrame)
secondView.backgroundColor = UIColor.green
view.addSubView(secondView)
}
}
```