



Objective-C Basics

Lab Exercise 1

Course Outcome:

CO1: Create iOS programs that will run on both iPhone and iPad devices following the Model-View-Controller design pattern in the iOS environment.

Exercise:

(LE11 xcodeproject)

Consider a scenario in which we observe the work of a teller in a bank. Each time a customer approaches the teller, she finds out what the customer would like. Some customers need change – for example, quarters for parking meters – while others may want to withdraw or deposit money. The teller then proceeds to either open her cash register or enter the customer's account number for a deposit or withdrawal.

Create some accounts using the following account numbers and balance amounts:

23457	\$5645.89
37678	\$564.67
56829	\$1235.35

Your program should welcome the customer and prompt the customer to enter one of the following numbers:

1. For deposit
2. For withdrawal
3. To check account balance
4. To get quarters for parking

Functions to create:

- Task selection that handles options 1 through 4.
 - `void selection(int num);`
- Deposit function should add the deposit amount to the account balance.
 - `void deposit(int account, double amt);`
- Withdrawal function deduct amount from the account balance.
 - `void withdrawal(int account, double amt);`
- Check Balance function that returns the account balance from a given account number.
 - `double checkBalance(int account);`
- Function to prompt the user for the amount to convert to change and print the number of quarters dispensed.
 - `void change(int num);`
- Get coins function that returns the number of quarters from amount given.
 - `int getCoins(double amt);`



Foundation Framework

Lab Exercise 2

Course Outcome:

C01: Create iOS programs that will run on both iPhone and iPad devices following the Model-View-Controller design pattern in the iOS environment.

Exercise:

(LE21 xcodeproject)

- 1) A teacher at an elementary school has organized an after-school recreation class. He asks you to create a program that will allow him to track which students attend the class. The class only accepts 10 students each day. Use an array to store the names and print them alphabetically.

(LE22 xcodeproject)

- 2) Revise the program from Programming Exercise 21 so that, in addition to storing the student's name, the program also stores the activity each student participates in. Assume that each student's name will be unique and use an NSDictionary object in your solution. The possible activities are:
 - basketball
 - floor hockey
 - arts
 - checkers



Object-Oriented Programming

Lab Exercise 3

Course Outcome:

CO1: Create iOS programs that will run on both iPhone and iPad devices following the Model-View-Controller design pattern in the iOS environment.

Exercise:

(LE31 xcodeproject)

- 1) A bookstore just opened and needs polo shirts for its employees. Each shirt should be display a quote from a famous author. The following are the quotes
- After a storm comes a calm.
 - Be kind whenever possible. It is always possible.
 - Have faith in your abilities!

The quote can be located on the front or back of the shirt. For colors, the employee can choose between red, green, black and blue. The shirts come in small, medium and large. Create a program that an employee can use to design his or her shirt. The program should allow user to enter his or her name and gender, select the shirt color, and enter desired quote, its location (whether it should be printed at the back or front), and size. The system should store the order in array. At the end of the day, the system should send all information entered that day to the print shop so that shirts can be created. Write a program that accomplishes this task using the principles of inheritance. The information sent to print shop should look like this:

Total Number of Orders =

Order Summary

Order Number	Name	Gender	Color	Size	Quote	Location
--------------	------	--------	-------	------	-------	----------



iOS Programming: UIControl Class

Lab Exercise 4

Course Outcome:

CO1: Create iOS programs that will run on both iPhone and iPad devices following the Model-View-Controller design pattern in the iOS environment.

Exercise:

(LE41 xcodeproject)

1) Create a number display app that includes the following:

- Two text fields in which the user can enter numbers
- Four buttons containing the text *Addition*, *Subtraction*, *Multiplication* and *Division*.
- One label that will display the result of the calculations.

When the user clicks any button, the result should be displayed in the label. In case any calculation errors (if user fails to enter one of the numbers), the label should display the word ERROR.

(LE42 xcodeproject)

2) Create a quote-of-the-day app. This app should contain at least one label and one button. Each time that the button is clicked, a new quote should appear in the label. Create an NSArray object that contains 10 quotes. Ensure that the same quote cannot appear consecutively (if user clicks on the button and gets the quote, the next time the user clicks the button, he or she should not get the same quote).

(LE43 xcodeproject)

3) Create a BMI calculator app with the following requirements:

Input:

- Label – Title
- Text Field – W in kg (Numeric Pad)
- Text Field – H in m (Numeric Pad)

Output:

- Text Field – BMI Factor
 - Label – BMI Evaluation (text)
 - Image View – BMI Evaluation (image)
- *displays value @ 2 decimal places

Event:

- Button – Evaluate BMI

Solve for BMI:

BMI Factor = W/H^2

< 18.5 = Underweight

18.5 – 24.9 = Normal

25 – 30 = Overweight

> 30 = Obese



Storyboard and Passing Data Between Views

Lab Exercise 5

Course Outcome:

CO1: Create iOS programs that will run on both iPhone and iPad devices following the Model-View-Controller design pattern in the iOS environment.

Exercise:

(LE51 xcodeproject)

- 1) Create a gift-ideas app. The app should contain three views. The first view should contain segmented control button for: Male and Female. The second view should contain segmented control button for: Infant, Adolescent and Adult. The third view should display the gift suggestions listed in the table:

First Screen	Second Screen	Suggestions
Male	Infant	toys trucks, blanket
Female	Infant	doll, blanket
Male	Adolescent	baseball jersey
Female	Adolescent	softball jersey
Male	Adult	tie, watch
Female	Adult	earrings, watch

(LE52 xcodeproject)

- 2) Mr. Picasso is an art instructor who wants to teach his students about mixing colors. Use storyboards to create an app that simulates the process of mixing paint to create a new color. The app should have three switches for the three primary colors (Red, Green and Blue) on its initial view. Also include a View Button. The user should be able to switch any of the three colors to ON or OFF and then click the View button to display a new view with its background set to match the new color that would result if you actually mixed the ON colors. Below is the summary of the color mixture:

RED	GREEN	BLUE		COLOR
OFF	OFF	OFF	=	Black
OFF	OFF	ON	=	Blue
OFF	ON	OFF	=	Green
OFF	ON	ON	=	Cyan
ON	OFF	OFF	=	Red
ON	OFF	ON	=	Magenta
ON	ON	OFF	=	Yellow
ON	ON	ON	=	White



Delegate and Protocol

Lab Exercise 6

Course Outcome:

CO1: Create iOS programs that will run on both iPhone and iPad devices following the Model-View-Controller design pattern in the iOS environment.

Exercise:

(LE61 xcodeproject) --- by PAIR

1) **Create a list app.** The app should contain 2 views embed in navigation controller. The first view should contain the table view controller where options are listed out and the second view should contain the view controller where detail of the choice is displayed. Make use of the table view delegate and protocol. The app should implement the following specification:

- Table View Controller
 - i. At least 3 sections
 - ii. At least 5 items per section
 - iii. Incorporate image on the cell/list
- Alert View for notification
- Sound (AudioToolbox) – 3 seconds or less sound file

(LE62 xcodeproject)

2) **Create and control a web view app.** The app should implement using web view, toolbars, action sheet and Spinner. Make use of the web view delegate and protocol. The app should implement the following specification:

- Toolbar – prompt Action Sheets
- Action Sheet – website options (at least 5 websites)
- Spinners – web view loading animation
- Web View – Display the web site selected



Mobile App Project

Final Course Requirement

Course Outcome:

CO2: Present a significant iOS business application based on real-world problems.

CO3: Participate in the planning and implementation of team-related analysis and design outputs.

Instructions:

- 1) Each team should have at most 5 members.
- 2) Group must come up with a significant business application using mobile iOS platform.
- 3) The mobile application should have at least the following minimum requirements:
 - a. Segues (Multi-scene)
 - b. Embed in Tab Bar / Navigation View Controller
 - c. Appropriate User Interfaces
 - d. Alert View / Action View
 - e. Images and Sounds
 - f. Table Views / Picker View / Web View (better option, as per app requires)
 - g. Dynamic Data Persistence: Property List / SQLite (better option, as per app requires)
 - h. App Icon
- 4) The group shall create a document based on the template provided at Google shared drive.
- 5) Actual checking and presentation of the mobile application project is required.
- 6) Submit the following through email:
 - a. Per group
 - i. Mobile App Project Document in pdf format.
 - ii. Compressed xcodeproject of the mobile app.
 - b. Individually
 - i. Self and peer evaluation according to the rubrics and scoresheet provided at Google shared drive