

Name: Khushi Nitinkumar Patel

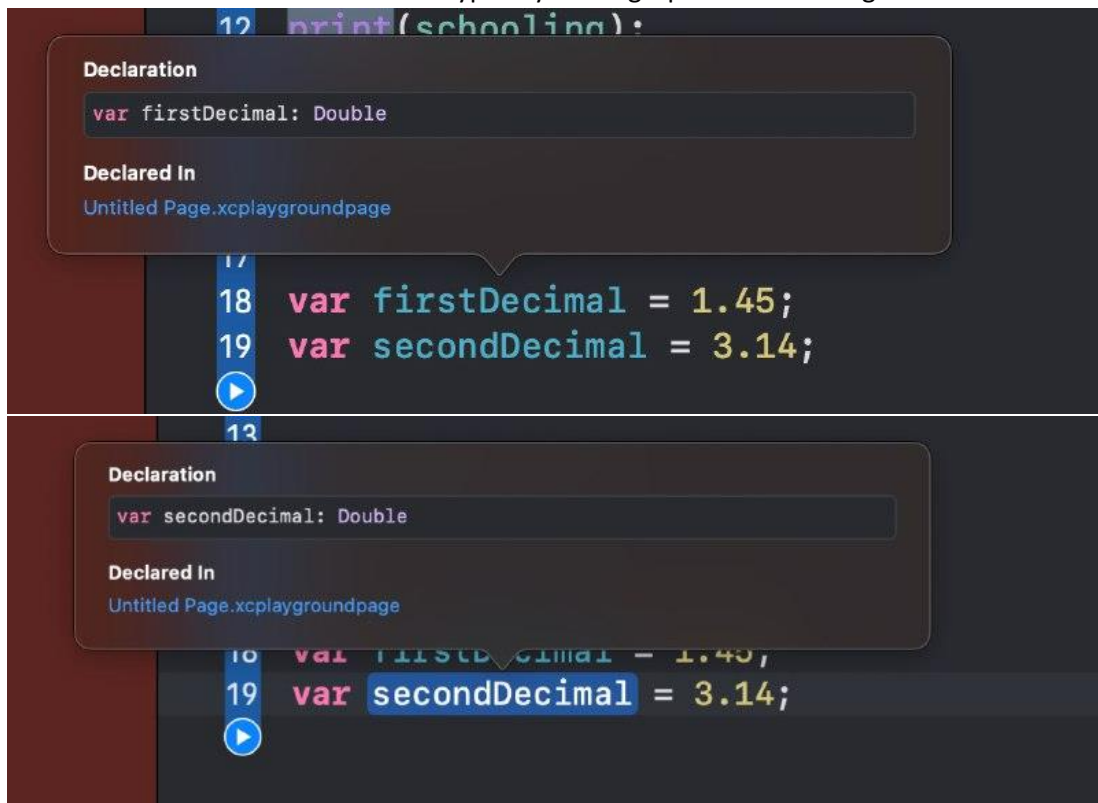
PRN: 2020BTECS00037

Batch: T2

Assignment No 2

Types and Type Safety

1. Declare two variables, one called `firstDecimal` and one called `secondDecimal`. Both should have decimal values. Look at both of their types by holding Option and clicking the variable name.



2. Declare a variable called `trueOrFalse` and give it a boolean value. Try to assign it to `firstDecimal` like so: `firstDecimal = trueOrFalse`. Does it compile? Print a statement to the console explaining why not, and remove the line of code that will not compile.

```
21 //2
22 var trueOrFalse = true;
23
24 firstDecimal = trueOrFalse;
```

Cannot assign value of type 'Bool' to type 'Double'

```
21 //2
22 var trueOrFalse = true;
23
24 //firstDecimal = trueOrFalse;
25 print("No the above statement doesn't compile. This happens
    because, each constant or variable in Swift has a type
    that describes its kind of value.")
```

☐ true
☐ "No the above statement doesn't comp..."

4 lines

No the above statement doesn't compile. This happens because, each constant or variable in Swift has a type that describes its kind of value.

3. Declare a variable and give it a string value. Then try to assign it to `firstDecimal`. Does it compile? Print a statement to the console explaining why not, and remove the line of code that will not compile.

```
26
27 //3
28
29 var str = "Khushi"
30 firstDecimal = str
31 print(firstDecimal)
```

Cannot assign value of type 'String' to type 'Double'

expression failed to parse:
error: Untitled Page.xcplaygroundpage:30:16: error: cannot assign value of type 'String' to type 'Double'
firstDecimal = str

```
29 var str = "Khushi"
30 //firstDecimal = str
31 //print(firstDecimal)
32 print("No the above statement doesn't compile because, once you assign a
    value to a constant or variable, the type is set and can't be
    changed. This is true even for variables. The value of a variable
    may change, but not its type.")
```

☐ "Khushi"
☐ "No the above statement doesn't compile because, onc..."

Line: 32 Col: 232

No the above statement doesn't compile because, once you assign a value to a constant or variable, the type is set and can't be changed. This is true even for variables. The value of a variable may change, but not its type.

4. Finally, declare a variable with a whole number value. Then try to assign it to `firstDecimal`. Why won't this compile even though both variables are numbers? Print a statement to the console explaining why not, and remove the line of code that will not compile.

```
33
34 //4
35 var x = 7
36 firstDecimal = x
37 print(firstDecimal)
```

Cannot assign value of type 'Int' to type 'Double'

expression failed to parse:
error: Untitled Page.xcplaygroundpage:36:16: error: cannot assign value of type 'Int' to type 'Double'
firstDecimal = x
 ^
 Double()

```
34 //4
35 var x = 7
36 //firstDecimal = x
37 //print(firstDecimal)
38 print("In the case above, both variables are numeric types, but  
      wholeNumber will be an Int and numberWithDecimals will be a Double.  
      In Swift, you can't assign a value of one type to a variable of  
      another type.")
```

7

"In the case above, both variables are numeric type..."

In the case above, both variables are numeric types, but wholeNumber will be an Int and numberWithDecimals will be a Double. In Swift, you can't assign a value of one type to a variable of another type.

5. You have declared a number of constants and variables to keep track of fitness information. Declare one more variable with a boolean value called `hasMetStepGoal`.

```
39
40 //5
41
42 let name = "Khushi"
43 let StepGoal = 2000
44 var noOfSteps = 1000
45 var hasMetStepGoal = false
```

6. When you declared a constant for goal number of steps and a variable for current step count, you likely assigned each a value in the thousands. This can be difficult to read. Redeclare this constant and variable and, when assigning each a value in the thousands, format the number so that it is more readable.

```
46
47 //6
48 let name = "Khushi"
49 let StepGoal = 2_000
50 var noOfSteps = 1_000
51 var hasMetStepGoal = false
```

Type Inference and Required Values

7. Declare a variable called `name` of type `String`, but do not give it a value. Print `name` to the console. Does the code compile? Remove any code that will not compile.

```
52
53 //7
54 var name:String
55 print(name)
```

Variable 'name' used before being initialized

expression failed to parse:
error: Untitled Page.xcplaygroundpage:55:7: error: variable 'name' used before being initialized
print(name)
^

Untitled Page.xcplaygroundpage:54:5: note: variable defined here
var name:String
^

8. Declare a variable called `distanceTraveled` and set it to 0. Do not give it an explicit type.

```
56
57 //8
58
59 var distanceTraveled = 0
```

9. Now assign a value of 54.3 to `distanceTraveled`. Does the code compile? Go back and set an explicit type on `distanceTraveled` so the code will compile.

```
61 //9
62 distanceTraveled = 54.3
63 print(distanceTraveled)
```

Cannot assign value of type 'Double' to type 'Int'

expression failed to parse:
error: Untitled Page.xcplaygroundpage:62:20: error: cannot assign value of type 'Double' to type 'Int'
distanceTraveled = 54.3
 ^~~~~
 Int()

```
59 var distanceTraveled = 0
60
61 //9
62 distanceTraveled = Int(54.3)
63 print(distanceTraveled)
```

54

10. You decide that your fitness tracking app should show the user what percentage of his/her goal has been achieved so far today. Declare a variable called `percentCompleted` and set it to 0. Do not explicitly assign it a type.

```
64
65 //10
66
67 var percentCompleted = 0
```

11. Imagine that partway through the day a user has taken 3,467 steps out of the 10,000 step goal. This means he/she is 34.67% of the way to his/her goal. Assign 34.67 to `percentCompleted`. Does the code compile? Go back and explicitly assign a type to `percentCompleted` that will allow the code to compile.

```
66
67 var percentCompleted = 0
68 percentCompleted = 34.67
69 print(percentCompleted)
```

Cannot assign value of type 'Double' to type 'Int'

expression failed to parse:
error: Untitled Page.xcplaygroundpage:68:20: error: cannot assign value of type 'Double' to type 'Int'
percentCompleted = 34.67
 ^~~~~~
 Int()

```
66
67 var percentCompleted:Double
68 percentCompleted = 34.67
69 print(percentCompleted)
```

34.67

Application design workbook

1. Explore your users

Explore Your Users

Pick one of the challenges and gather information about individuals who experience it. Each person is different. It's important to think broadly to capture as much diversity as you can.

Good design is user-centered. You've gotten a good start by thinking about the challenges that you and others face. Keep it going! By narrowing down from the general to the specific, you'll place individual people at the core of your process.

Personal stories from real people can give you perspective you might not otherwise have. Consider interviewing people from your community to create authentic profiles.

Define | Discover

Example

Who is this person? How do they describe themselves?

A father of two, kindergarten teacher, taking online classes in photography.

How old are they?

31

What are important aspects of their environment?

Live in an apartment on the third floor. Not enough space for large bins in the house, so they only have a small recycling container.

How do they describe the challenge they face?

I don't really understand what's recyclable and what belongs in the trash. The labels are hard to find, and I don't really how to distinguish between things like different kinds of paper.

What do they most want in a solution? How would it make their lives easier?

I need help quickly identifying what's recyclable. If I could sort through items quickly every evening, I'd be more likely to spend mental energy on it, since my kids deserve as much energy as I can give them.

In which specific circumstances might they use an app that addresses their challenge?

I could spend a little time every evening with my kids sorting through our daily waste.

2. Consider diversity

Consider Diversity


Identify things about your users you may have overlooked.

A user's identity and circumstances will have a huge impact on how they'll experience and use an app. Summarize all your users with these different aspects in mind.

Everyone has biases that affect the way they perceive the world. Compensate for your biases so that they don't creep into your app's design.

Did you identify something that you didn't consider when imagining your audience? For example, were all your users of similar age? Consider going back to the earlier exercises with your new insights in mind.

Define | Discover



Ages

Above 18

Genders

All

Languages

English, Hindi and Marathi

Disabilities


NA

Cultures

NA

Economic circumstances

Living situations




3. Summarize your audience

Summarize Your Audience

Summarize your findings about individual users. Refer to your earlier research and use it to draw some conclusions.

Define | Discover



Example

What's the most important concern in a solution?

Understanding the percentage of trash vs. recycling.

The age range of the users is:

15 to 30.

Our app will be opened when...

Throwing things in the trash or recycling.

Our app will be used in this environment:


Inside with connection to Wi-Fi or cell reception.

Our environment will have these limitations:

User may have their hands full.

When designing our app, we need to consider:

Users might not know what qualifies as a recyclable.



4. Analyze causes

Analyze Causes

Dig deeper into the issues you've observed and find the core problem. Then consider how your app could solve it.

Asking why something happens will help you discover hidden causes behind what you observe directly. The deeper you dig, the closer you'll get to the core motivating need for your solution.

Create as many copies of the following template as you need to describe the problems you've identified in your research.

Define | Analyze



Example

Users are having this problem:

① They'd like to begin recycling but struggle to hold themselves accountable.

↓

This happens because:

② It's easier to throw everything into one bin.

↓

This is because:

③ They struggle to differentiate between trash and recycling.

↓

Which is because:

④ Recycling seems complicated and hard to remember.

↓

And the root cause is this core problem:

⑤ People are rarely taught how to recycle.

We can solve this issue in our app by:

Educating people on what qualifies as recycling and gamifying the experience so they can hold themselves accountable with their peers.

5. Research Competitors

Research Competitors

Find and describe apps that relate to the problem you've identified.

Discover what people are currently using to solve the problem. Search the App Store for similar apps to find out what users enjoy or dislike about their solution. This will give you insight into what your app will be competing with.

Define | Analyze



	This app is interesting because:	I like/dislike this app because:
	Easy to use	Only shows expenses
	Provides daily expense tracker	No visual charts available
	GUI	Lots of advertisements
	Recommendation for daily input	Unable to provide expense and saving for certain period
	Easy to create balance sheet at the end of month	Unable to attach proof

Explore your users

Who is the person? How do they describe themselves?

An ordinary person who likes listening to music.

How old are they?

20

What are important aspects of their environment?

Likes listening to music but cannot find a proper playlists of songs to listen to according to their mood and language preference.

How do they describe the challenge they face?

I can't find a playlist that I would like to listen to, depending upon my mood.

What do they most want in the solution? How would it make their lives easier?

I need multiple options of a person's mood and after clicking the option I would be redirected to a playlist containing all the songs suitable to my mood and according to my language preference.

In which specific circumstances might they use an app that addresses their challenge?

After an exhausting day of work I can listen to a playlist of songs depending upon my mood.

Consider diversity

1. **Ages:** 18 and above.
2. **Languages:** No language barrier
3. **Disabilities:** NA
4. **Cultures:** NA
5. **Economic circumstances:**
6. **Living situations:**

Summarize your audience

What's the most important concern in a solution?

Understanding the taste of the user in music according to his/her mood.

The age range of the users is:

18 and above

Our app will be opened when..

User wants to play a particular playlist of songs that can help him/her to handle their mood/emotions.

Our app will be used in this environment:

Everywhere with connection to Wi-Fi or Internet data.

Our environment will have these limitations:

NA

When designing our app we need to consider:

Users might not know about the playlist that can handle their current emotions.

Analyse causes

Users are having this problem:

They'd like to listen to a playlist of songs depending on their mood but cannot find one.

This happens because:

It's difficult to find a playlist related to their current mood.

This is because

They struggle to find songs related to their current mood.

Which is because

Sorting songs in a playlist of a particular genre or feeling or emotion seems time consuming.

And the root cause is this core problem:

People need quick access to a playlist which can support their mood.

We can solve this issue in our app by:

Creating multiple options of user's emotions for example: sad, happy, angry, lonely, etc. Which can give quick access to a playlist of user's choice of mood.