Slide 1

Hamza:

My name is Hamza

Ahmed:

My name is Ahmed. Our project focuses on Swift

Slide 2

Hamza:

Swift is Apple's programming language exclusively for Apple products.
It was originally developed by U of I grad, Chris Lattner and
was publicly released in June of 2014.
Before Swift, all Apple applications were developed using Objective-C.

Slide 3

Ahmed:

Swift is mainly an object-oriented language, but it could also be categorized as imperative because of its use of mutable objects, and functional since it contains reduce/map-like functions.
(explain pictures)

Slide 4

Hamza:

Swift is known to be a relatively easy to learn language, even without any prior coding experience. Apple encourages non-programmers to dabble with programming since they released an iPad app that helps teach programming to a younger audience.

Swift is used to develop all Apple applications
(Mac, iPhone, iPad, Watch OS, TV OS)

Xcode is Apple's free IDE where Swift apps are best developed. Playground is an intersting feature included in Xcode that outputs the program while you write it. The picture is from an Apple event where the playground feature was first introduced.

Slide 5

Ahmed:

The key feature that we found most intriguing is Swift's use of 'weak' and 'strong' references.
A strong reference is an instance that has a strong link to another object so that it can't be destroyed.
If compared to Java, all objects in Java are passed by reference and won't be garbage collected until it isn't referenced.
In Swift all objects are strong by default, but a programmer can make the object weak if needed.

All instances are deallocated by the ARC if no other objects have a strong reference to it.

It isn't easy to explain in words, so we designed our demo to show the difference between a weak and strong reference.

Slide 6

demo

step 1:

explain the Student and Grade class

As you can see we have a student class and a grade class. The student class a public variable which takes in a name. We then create a initializer which will accept the name. We also have a deinitializer which will delete the name.

We also have created a grade class which basically does the same thing as a grade class but takes in an int.

step 2:

uncomment main code to run default code

step 3:

a. comment default classes
b. uncomment second classes

c. comment line in main
d. uncomment line in main
two strong references

We add a new array to the Student class which will hold the grades for a particular student. We also add a property to the Grade class and then we modify that class so that it accepts it.

step 4:

run main code to show objects aren't able to be destroyed

The objects we created have not been destroyed which basically means that there is a memory leak.

step 5:

create a weak reference
a. comment out line 80
b. uncomment line 81

step 6:

run main code

Slide 7

Hamza:

Here's a more visual explanation of what was going on with the 2 strong references.

The grade object first creates a strong reference in the first line in the red circle. Then, the array of grades in the student object creates a strong reference to the actual grade object in the second line. Since both of them have a strong reference on each other, the ARC can't garbage collect them which becomes a memory leak, which is called a strong reference cycle.

Slide 8

Ahmed:

As we said earlier, the language that was used to make all Apple applications before Swift was Objective-C. Swift was created as a means to improve the C-like confusing looking code that was Objective-C. Most programmers developing code for Apple products will more likely choose Swift over Objective-C because Swift requires less code which makes it much more readable. Swift is also safer and less error-prone for similar reasons of C++ being safer than C; Less memory management by the user, less raw pointers, and better handling of NULL/nil exceptions.

The code on the left and right are Objective-C and Swift performing the same tasks, respectively.

The Swift section is much easier to decipher than the Objective-C section.

Slide 9

Hamza:

Apple always make a heavy emphasis on safety in all of their products. That is also the case with Swift. Swift is much more of a safer language than Objective C. It automatically allocates and deallocates memory for you and is much more easier to read than Python or Objective C.

The downside to Swift is that it is apple exclusive. It can only be used on a machine running the OS X. Another one is that Swift is a relatively new language which is why every update that comes out for it changes a lot from previous versions. It is already in its 4th major update which shows how much the language is still developing.

Slide 10

Ahmed:

Here are a few statistics we found online that we thought were pretty interesting. According to StackOverflow's 2017 Developer Survey, Swift ranked as the 4th 'Most Loved' language, and 8th as 'Most Wanted.'

Slide 11

Hamza

And according to the same survey, Swift didn't even make it onto the list of 'Most Dreaded' language.