

UNIVERSITY

COURSE NAME

Title of Report

Authors

Henry Terry hter@example.com
Amy Shelton ashe@example.com

Supervisor

Lucas Watts lwat@example.com

November 30, 2016

Table of Contents

1	Dynamic languages	3
1.1	Ruby	3
1.2	Python	3
1.3	JavaScript	3
2	Static languages	5
2.1	Java	5
2.2	C#	5
3	Functional languages	6
3.1	Haskell	6
3.2	F#	6

1 Dynamic languages

1.1 Ruby

Ruby is a dynamic, reflective, object-oriented, general-purpose programming language.

Code examples

Hello World

```
puts "Hello World!"
```

String interpolation

```
var = 3.14159  
"pi is #{var}"  
=> "pi is 3.14159"
```

1.2 Python

Python is a widely used general-purpose, high-level programming language.

```
print "Hello World!"
```

1.3 JavaScript

JavaScript, also known as ECMAScript (the untrademarked name used for the standard), is a dynamic programming language.

```
console.log('Hello World!');
```

2 Static languages

2.1 Java

Java is a general-purpose computer programming language that is concurrent, class-based, object-oriented, and specifically designed to have as few implementation dependencies as possible.

```
class HelloWorldApp {  
    public static void main(String[] args) {  
        System.out.println("Hello World!");  
    }  
}
```

2.2 C#

C# is a multi-paradigm programming language encompassing strong typing, imperative, declarative, functional, generic, object-oriented (class-based), and component-oriented programming disciplines.

```
using System;  
  
class HelloWorldApp  
{  
    static void Main()  
    {  
        Console.WriteLine("Hello World!");  
    }  
}
```

3 Functional languages

3.1 Haskell

Haskell is a standardized, general-purpose purely functional programming language, with non-strict semantics and strong static typing.

```
module Main where

main :: IO ()
main = putStrLn "Hello, World!"
```

3.2 F#

F# is a strongly typed, multi-paradigm programming language that encompasses functional, imperative, and object-oriented programming techniques.

```
printfn "Hello World!"
```