University

Course Name

Title of Report

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

Supervisor
Lucas Watts lwat@example.com

October 19, 2016

Table of Contents

1 Dynamic languages

1.1 Ruby

Ruby is a dynamic, reflective, object-oriented, general-purpose programming language (Freely 1997).

Code examples

Some simple code examples (1997).

Hello World

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

1.2 Python

Python is a widely used general-purpose, high-level programming language.[1]

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

1.3 JavaScript

JavaScript, also known as ECMAScript (the untrademarked name used for the standard), is a dynamic programming language. See Doe (1337).

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

[1] Jass (7991)

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.

```haskell
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.

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

Bibliography

Doe, John. 1337. “A Medium Paper.” *Some Medium Papers*.

Freely, I.P. 1997. “A Small Paper.” *The Journal of Small Papers*.

Jass, Hugh. 7991. “A Big Paper.” *The Journal of Big Papers* MCMXCVII.