Before we get going.   Lets have everyone quickly introduce themselves and in a few words, why did you join.  What do you want to take away from this class and the upcomming classes?

Lets talk about technology briefly, developers, how I will conduct these sessions and how I see becoming a truly great developer.

-----------------

Technology :

Thank you wheel

-----------------

Great Developers:

What makes a great developer instead of a good developer.  A good developer will talk your ear off about syntax and latest tools.  But they will never share with you great tools they wrote in a language they only just learned.  Good developers will rarely mentor you in a way you feel like you could not have learned it without them.  Great developers will mentor you, they will share the code for things you cant solve or ways you should consider.  Good developers can only change their own code with ease.  Great developers can make anyones code better.

Great developers practice these habits

1. Positivity and almost a silly nature to deal with stress.

2. They communicate what they know openly and clearly for others

3. They learn quickly

4. They know about all of IT not just their area

5. They are not language or platform biggots

6. They know the team is more important than the single individual even if they silently put in the most time

7. Their programs are creations, not just efforts

8. Every time they look at their code they realize they could have done it better

9. QA is a checkbox expected to be checked because they already tested all possible options.

10. They depend exclusively on debug to know what a program did wrong

-----------------

Success:  I love a quote I read from a Kobe Bryant interview.

Among many other things he said he summarized how I live and have lived since I was 19 and decided to be a software engineer.

"

I knew that I was not going to be stopped... you can't possibly become better than me, because you're not spending the time on it that I do.

Even if you want to spend the time on it, you can't, because you have other things. You have other responsibilities taking you away from it. So I already won."

--------------------

Format of information:

Go over directories

       Installs

       Notes

       Examples

       Code

---------------------

Help:

Call me, email me, text me or

1. Goals

    a. Not to create senior developers, but instead ensure more people are comfortable with languages and how to learn them

    b. Create a way of thinking about development that creates quality, ensures everyone can help.

    c. Discuss development in the technology space

         ca.  Thank you wheel

         cb.  Software and bugs

         cc.  Procedural

                 Scripting

                 Object Oriented

                 Full stack

                 whats next

    d. Discuss what we will build and how rapidly we will make complete applications

         da. Speed of learning

         db. Frequency of collaboration

         dc. Smaller groups as working sessions

         dd. Do most exercises in multiple languages to see how they are similar and relate

    e. Discuss platforms, versions, etc

         ea. OS - Windows v Mac v Linux

         eb. DB - Sql v MySQL v Oracle v Mongo

         ec. Language - Java v C# v Node v Python v Java Script

         ed. Editors or IDE - VS v VS code v Jetbrains v Eclipse v VI or Notepad++

    f. Start our first class

Python vs C#

**Python is an Interpreted**− Python that is processed at runtime by the interpreter. Before you run it you don’t have to compile your program. This is often kind of like PERL and PHP.

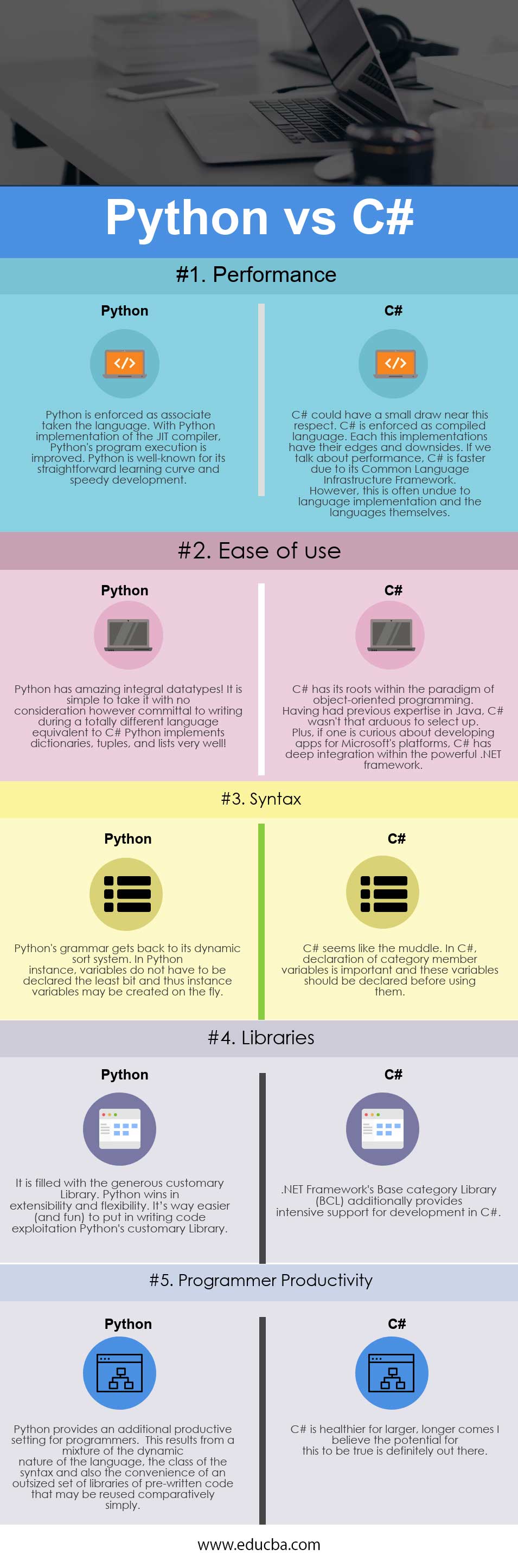
**Python is Interactive −** you’ll truly sit at a Python prompt and act with the interpreter to write your programs.

**Python is Object-Oriented −** Python supports programming technique in the Object-Oriented programming style. Within objects, Python encapsulates code.

**Python is a Beginner’s Language −** Python is a nice language for the beginner-level programmers and supports the event of a good variety of applications from straightforward text process to World Wide Web browsers to games.

The following reasons build C# a wide used skilled language −

* It is a fashionable, all-purpose artificial language
* It is object orientated.
* It is part orientated.
* It is simple to be told.
* It is a structured language.
* It produces economic programs.
* It may be compiled on a range of pc platforms.
* It is a part of.Net the Framework.



Node vs ASP.NET

Lets dive into the specific differences between the two and see if we can conclusively decide which one of them is a better option.

1) **Language**: ASP.NET makes use of C# as the primary language for development work. Whereas, Node.js uses JavaScript. The advantage of C# is that it provides a stringent type system along with compile-time error checks. And let's not forget that C# is renowned for becoming more expressive and efficacious with every new iteration. On the other hand, JavaScript, which depends on Facebook’s Flow static type checker or Microsoft TypeScript, seems to be currently more favored as compared to C#. However, developers who choose Node.js must first master asynchronous programming, which, by the way, is no easy task.

2) **The simplicity of development**: As compared to ASP.NET, Node.js offers fewer abstractions. It, in turn, empowers developers to write code with several small components instead of setting up numerous parameters. As a result, Node.js is much more agile when it comes to customizing the code by the solution. Another benefit of Node.js is that it provides native as well as third-party libraries.

However, ASP.NET gets various conventions that allow developers to ensure that their code is not only brief but also readable. Unfortunately, deviating from these rules means they have to use manual configuration, which results in an increased workload for the developer.

3) **Scalability**: When it comes to the microservices architecture strategy, which delivers superior stability and scalability, Node.js is the ideal choice since its elements are compartmentalized. Furthermore, Node.js tools like Rollbar and the PM2 process manager help identify and monitor errors much more easily.

ASP.NET too has much to offer in the context of scalability -- the framework facilitates the efficient analysis of an extensive amount of user data. Another advantage with ASP.NET is that its scaling-up process requires fewer additional machines than one might presume.

Procedural Vs Object Oriented

| **Procedural Oriented Programming** | **Object Oriented Programming** |
| --- | --- |
| In procedural programming, program is divided into small parts called **functions**. | In object oriented programming, program is divided into small parts called **objects**. |
| Procedural programming follows **top down approach**. | Object oriented programming follows **bottom up approach**. |
| There is no access specifier in procedural programming. | Object oriented programming have access specifiers like private, public, protected etc. |
| Adding new data and function is not easy. | Adding new data and function is easy. |
| Procedural programming does not have any proper way for hiding data so it is **less secure**. | Object oriented programming provides data hiding so it is **more secure**. |
| In procedural programming, overloading is not possible. | Overloading is possible in object oriented programming. |
| In procedural programming, function is more important than data. | In object oriented programming, data is more important than function. |
| Procedural programming is based on **unreal world**. | Object oriented programming is based on **real world**. |
| Examples: C, FORTRAN, Pascal, Basic etc. | Examples: C++, Java, Python, C# etc. |

Developer Types

• front end development (the visible parts of a website or app)

• back end development (the “under the hood” databases and infrastructure)

• full stack development (a hybrid of both). Full stack can apply to a web stack, mobile stack, or a native application stack (i.e. software programs for specific devices)

Mongo Vs MySQL

The Major Differences between MongoDB and MySQL

1. There is a difference in the representation of data in the two databases. In MongoDB, data represents in a collection of JSON documents while in MySQL, data is in tables and rows. JSON documents can compare to associative arrays when using PHP and directory objects when using Python.

2. When it comes to querying, you have to put a string in the query language that the DB system parses. The query language is called Structured Query Language, or SQL,from where MySQL gets its name. This exposes your DB susceptible to SQL injectionattacks. On the other hand, MongoDB’s querying is object-oriented, which means you pass MongoDB a document explaining what you are querying. There is no parsing whatsoever, which will take some time getting used to if you already use SQL.

3. One of the greatest benefits of relational databases like MySQL is the JOIN operation. The operation allows for the querying across several tables. Although MongoDB doesn’t support joints, it supports multi-dimensional data types like other documents and arrays.

4. With MySQL, you can have one document inside another (embedding). You would have to create one table for comments and another for posts if you are using MySQL to create a blog. In MongoDB, you will only have one array of comments and one collection of posts within a post.

5. MySQL supports atomic transactions. You can have several operations within a transaction and you can roll back as if you have a single operation. There is no support for transactions in MongoDB and the single operation is atomic.

6. One of the best things about MongoDB is that you are not responsible for defining the schema. All you need to do is drop in documents. Any 2 documents in a collection need not be in the same field. You have to define the tables and columns before storage in MySQL. All rows in a table share the same columns.

7. MongoDB’s performance is better than that of MySQL and other relational DBs. This is because MongoDB sacrifices JOINS and other things and has excellent performance analysis tools. Note that you still have to index the data and the data in most applications is not enough for them to see a difference. MySQL is criticized for poor performance, especially in ORM application. However, you are unlikely to have an issue if you do proper data indexing and you are using a database wrapper.

8. One advantage of MySQL over NoSQL like MongoDB is that the community in MySQL is much better than NoSQL. This is mostly because NoSQL is relatively new while MySQL has been around for several years.

9. There are no reporting tools with MongoDB, meaning performance testing and analysis is not always possible. With MySQL, you can get several reporting tools that help you rove the validity of your applications.

10. RDBSs function on a paradigm called ACID, which is an acronym for (Atomicity, Consistency, Isolation, and Durability). This is not present in MongoDB database.

11. MongoDB has a Map Reduce feature that allows for easier scalability. This means you can get the full functionality of MongoDB database even if you are using low-cost hardware.

12. You do not have to come up with a detailed DB model with MongoDB because of is non-relational. A DB architect can quickly create a DB without a fine-grained DB model, thereby saving on development time and cost.

| **MS SQL Server** | **MySQL** |
| --- | --- |
| Developed by Microsoft. | Developed by Oracle. |
| It supports programming languages like C++, JAVA, Ruby, Visual Basic, Delphi, R etc. | MySQL offers extended running support for languages like Perl, Tcl, Haskey etc. |
| Expects a large amount of operational storage space. | Expects less amount of operational storage space. |
| It enables for stopping query execution. | It doesn’t allow query cancellation mid-way in the process. |
| Doesn’t block the database while backing up the data. | Blocks the database while backing up the data. |
| It is not free. | It is open source. It is freely available. |
| It is a highly secured and doesn’t allow any kind of database file manipulation while running. | It allows database file manipulation while running. |
| It is available in multiple editions, such as Enterprise, Standard, Web, Workgroup, or Express. | It is available in MySQL Standard Edition, MySQL Enterprise Edition, and MySQL Cluster Grade Edition. |

Node Vs ASP.NET

Lets dive into the specific differences between the two and see if we can conclusively decide which one of them is a better option.

1) **Language**: ASP.NET makes use of C# as the primary language for development work. Whereas, Node.js uses JavaScript. The advantage of C# is that it provides a stringent type system along with compile-time error checks. And let's not forget that C# is renowned for becoming more expressive and efficacious with every new iteration. On the other hand, JavaScript, which depends on Facebook’s Flow static type checker or Microsoft TypeScript, seems to be currently more favored as compared to C#. However, developers who choose Node.js must first master asynchronous programming, which, by the way, is no easy task.

2) **The simplicity of development**: As compared to ASP.NET, Node.js offers fewer abstractions. It, in turn, empowers developers to write code with several small components instead of setting up numerous parameters. As a result, Node.js is much more agile when it comes to customizing the code by the solution. Another benefit of Node.js is that it provides native as well as third-party libraries.

However, ASP.NET gets various conventions that allow developers to ensure that their code is not only brief but also readable. Unfortunately, deviating from these rules means they have to use manual configuration, which results in an increased workload for the developer.

3) **Scalability**: When it comes to the microservices architecture strategy, which delivers superior stability and scalability, Node.js is the ideal choice since its elements are compartmentalized. Furthermore, Node.js tools like Rollbar and the PM2 process manager help identify and monitor errors much more easily.

ASP.NET too has much to offer in the context of scalability -- the framework facilitates the efficient analysis of an extensive amount of user data. Another advantage with ASP.NET is that its scaling-up process requires fewer additional machines than one might presume.

Training Links

[Blockchain Beginners](https://www.udemy.com/course/blockchain/learn/lecture/6026656#overview)

[Blockchain Fundamentals](https://www.pluralsight.com/courses/blockchain-fundamentals?clickid=1xnX1Y0FCxyOWPQwUx0Mo38XUkn1kYWEE3Q0y00&irgwc=1&mpid=1193463&utm_source=impactradius&utm_medium=digital_affiliate&utm_campaign=1193463&aid=7010a000001xAKZAA2)

[Blockchain – Principles and Practices](https://www.pluralsight.com/courses/blockchain-principles-practices?clickid=1xnX1Y0FCxyOWPQwUx0Mo38XUkn1kYQME3Q0y00&irgwc=1&mpid=1193463&utm_source=impactradius&utm_medium=digital_affiliate&utm_campaign=1193463&aid=7010a000001xAKZAA2)

[Courses Dashboard | Wes Bos](https://courses.wesbos.com/account)

[Machine Learning](https://developers.google.com/machine-learning/crash-course/ml-intro)

[Machine Learning Crash Course](https://developers.google.com/machine-learning/crash-course/ml-intro)

[Python GUI examples (Tkinter Tutorial) - Like Geeks](https://likegeeks.com/python-gui-examples-tkinter-tutorial/)

[Udemy](https://www.udemy.com/)

[A Cloud Guru](https://acloud.guru/members/welcome)

[Online Regex Tester](https://regex101.com/)

[Regex Buddy](https://www.regexbuddy.com/library.html)

[Upsource: Code Review, Project Analytics, and Team Collaboration by JetBrains](https://www.jetbrains.com/upsource/promo/?gclid=Cj0KCQiA4NTxBRDxARIsAHyp6gD6X_B-KtI5S-2HEWyGSKKQrpYmIyHwrLxBiqMlhcAQXN-Gf0SikCMaApPbEALw_wcB)

[MLab - Node Class](https://mlab.com/home?newLogin=1)

[Branching With GIT](https://www.atlassian.com/git/tutorials/comparing-workflows/gitflow-workflow)

[PostgreSQL command line cheatsheet](https://gist.github.com/Kartones/dd3ff5ec5ea238d4c546)

[HTML Code Editor - Instant Preview](https://htmlcodeeditor.com/)

[HTML-JSON-CSS Editor](https://htmlg.com/member/?id=1118397592178913476122639060323866)

[Online HTML Editor - 𝗛𝗧𝗠𝗟-𝗢𝗻𝗹𝗶𝗻𝗲.𝗰𝗼𝗺](https://html-online.com/editor/)

[Google Developers](https://developers.google.com/)