

Part two:

1.

The main differences between a data warehouse and a typical SQL database include their main utilizations. A data warehouse is used for storing, managing, and retrieving large amounts of data to analyze it. A SQL database is mainly used for creating, reading, updating, and deleting data (CRUD).

For example, an online ecommerce store can use a SQL database to deal with their daily sales and inventory but may use a data warehouse to analyze trends in sales over time.

Inmon, W. H. (2005, September 19). *Building the data warehouse*. Google Books.

https://books.google.com/books/about/Building_the_Data_Warehouse.html?id=QFKTmh5IFS4C

2.

Tracking expenditures in an Excel spreadsheet is better for smaller companies with simple data. Using a SQL database is better for larger companies with more complex data and data needs.

For example. Someone who does freelance work for a small company would utilize Excel to track expenditures, but a large company like Walmart would use a SQL database to track expenditures.

Google. (n.d.). *Database system concepts 6th edition.pdf*. Google Drive.

<https://drive.google.com/file/d/13q8K7vhvQaavlgO1eJpKQHKjaTBh8dP/view?usp=sharing>

3.

To produce an executable file, a programmer must write the code, then compile it, and then link the libraries and other references.

For example, for a programmer to write a Java file, they would need to write the code inside a .java file, then compile it with the Java compiler, and then link them together.

Prepbytes. (2023, December 26). *Java compilation process*. PrepBytes Blog.

<https://www.prepbytes.com/blog/java/java-compilation-process/#:~:text=The%20Java%20compiler%20takes%20the,bytecode%20and%20executes%20the%20program.>

4.

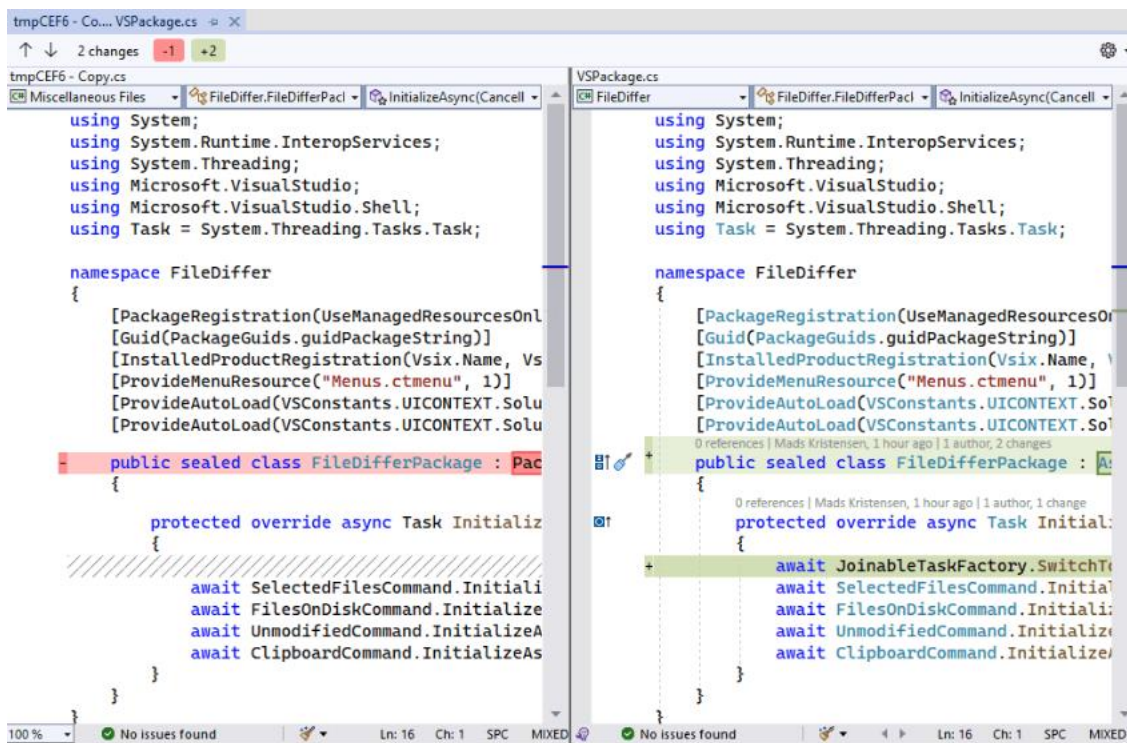
Version control systems are crucial in software projects because they manage the changes made to the project, allow multiple people to work on the same project in real time, and they track the history of all changes and show who made such changes.

There are many examples of alternatives to GitHub, but two are GitLab and Bitbucket. GitLab provides version control like GitHub but also has CI/CD pipelines. Bitbucket is similar to GitHub but is more private based and utilized by smaller companies.

Chacon, S., & Straub, B. (2014). **Pro Git** (2nd ed.). Apress

5.

A diff file is a text file that shows the difference between two versions of a file. Diff files are very helpful because they highlight changes such as added, removed, or edited code. This makes it very easy to spot differences and make changes.



The screenshot displays a side-by-side comparison of two versions of a C# file named `VSPackage.cs`. The left pane shows the original code, and the right pane shows the modified code. The diff highlights changes with color: red for deletions and green for additions. In the original code, the `FileDifferPackage` class has a `protected override async Task InitializeAsync()` method that calls `SelectedFilesCommand.InitializeAsync()`, `FilesOnDiskCommand.InitializeAsync()`, `UnmodifiedCommand.InitializeAsync()`, and `ClipboardCommand.InitializeAsync()`. In the modified code, the `InitializeAsync` method is replaced with `protected override async Task Initialize()`, which calls `JoinableTaskFactory.SwitchToMainThreadAsync()` before the other initialization calls. The status bar at the bottom indicates 'No issues found' and 'Ln: 16 Ch: 1 SPC MIXED'.

```
using System;
using System.Runtime.InteropServices;
using System.Threading;
using Microsoft.VisualStudio;
using Microsoft.VisualStudio.Shell;
using Task = System.Threading.Tasks.Task;

namespace FileDiffer
{
    [PackageRegistration(UseManagedResourcesOnly = true,
        [Guid(PackageGuids.guidPackageString))]
    [InstalledProductRegistration(Vsix.Name, Vsix.Name, 1)]
    [ProvideMenuResource("Menus.ctmenu", 1)]
    [ProvideAutoLoad(VSConstants.UICONTEXT.SolutionExplorer)]
    [ProvideAutoLoad(VSConstants.UICONTEXT.SolutionExplorer)]

    public sealed class FileDifferPackage : Package
    {
        protected override async Task InitializeAsync()
        {
            await SelectedFilesCommand.InitializeAsync();
            await FilesOnDiskCommand.InitializeAsync();
            await UnmodifiedCommand.InitializeAsync();
            await ClipboardCommand.InitializeAsync();
        }
    }
}
```

```
using System;
using System.Runtime.InteropServices;
using System.Threading;
using Microsoft.VisualStudio;
using Microsoft.VisualStudio.Shell;
using Task = System.Threading.Tasks.Task;

namespace FileDiffer
{
    [PackageRegistration(UseManagedResourcesOnly = true,
        [Guid(PackageGuids.guidPackageString))]
    [InstalledProductRegistration(Vsix.Name, Vsix.Name, 1)]
    [ProvideMenuResource("Menus.ctmenu", 1)]
    [ProvideAutoLoad(VSConstants.UICONTEXT.SolutionExplorer)]
    [ProvideAutoLoad(VSConstants.UICONTEXT.SolutionExplorer)]

    public sealed class FileDifferPackage : Package
    {
        protected override async Task Initialize()
        {
            await JoinableTaskFactory.SwitchToMainThreadAsync();
            await SelectedFilesCommand.InitializeAsync();
            await FilesOnDiskCommand.InitializeAsync();
            await UnmodifiedCommand.InitializeAsync();
            await ClipboardCommand.InitializeAsync();
        }
    }
}
```

Hunt, A., & Thomas, D. (2019). **Pragmatic Programmer, The: Your Journey To Mastery** (20th Anniversary Edition, 2nd ed.). Addison-Wesley

6.

A baseline is a fixed point of reference that is used to compare softwares. In a project, the baseline is typically the safest version of the project and is used as a foundation to develop the project even more.

McConnell, S. (2004). **Code Complete: A Practical Handbook of Software Construction** (2nd ed.). Microsoft Press.

7.

Program version in a private workspace is different from the baseline version because the program version is the most recent state of the project while the baseline version is the last release version from where the current changes are being made.

A conflict when two updates are made to the same part of the same file creates an error because they cannot be automatically merged. These conflicts require manual review and the programmers need to decide which change is better.

Fowler, M. (2004). **Refactoring: Improving the Design of Existing Code**. Addison-Wesley Professional.

8.

The build is the process of converting source code to software artifacts that can be executed on a computer. The result of the build is an executable application.

McConnell, S. (2004). **Code Complete: A Practical Handbook of Software Construction** (2nd ed.). Microsoft Press

9.

Three tier architecture is a software design that splits the project into three layers, including presentation, application, and data.

Martin, R. C. (2003). **Agile Software Development, Principles, Patterns, and Practices**. Prentice Hall.

10. polymorphism in technology is a concept that allows objects of one class to have the same properties even though its in a different class.

For example, when dealing with shapes, the square class could have an object for calculating area (length x height) and that object can be carried over to the rectangle class even though a rectangle isn't a square.

Gamma, E., Helm, R., Johnson, R., & Vlissides, J. (1994). *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley