# Fast Startup & Start Window

## Better Git Integration

* Git Clone direct from the url without opening the complete workspace of visual studio.
* Sign in to you github account inside of visual studio and it’ll show you the repos. Select & download them bam!

## Better Template Searching

* Filters
* Narrow down searches
* Smarter window
* Support for multiple languages (python etc.)
* Frequently used templates on the left panel in start window

# Live Share

* Share content instead of pixels
* Cursor shares from multiple location
* Debugging
* Localhost is shared across the users
* intelliSense also works!
* Full context development across remote access

# IntelliCode

* Instead of chronological order. Let’s be smart!
* Recommended suggestions (based on thousands of GitRepos & code of community Microsoft did AI & ML to these codes to provide best results)
* Most likely functionality will come on the top
* Everyone doesn’t like the same flavor of IntelliCode. So you can TRAIN your IntelliCode.
* Sharing of public & private wisdom for private projects across teams.
* Intelligent to identify the contexts. (Class, function, loop, blocks, etc.) First class citizens for intelliCode in VS19 are:

1. C#
2. Xaml
3. C++

Where as in VSCode its for:

1. Typescript
2. Javascript
3. Python
4. Java

# Useful Links

* Visualstudio.com/downloads – Community
* My.visualstudio.com - Goodies for Visual Studio Professionals & Enterprise subscriber
* Launch.visualstudio.com – Visual Studio 2019 technical training courses on PluralSight & LinkedIn Learning

# Delight Feature

## Data breakpoints

* You can now break when an object value changes in C#. This was available to C++ developers but now to C# as well. It doesn’t matter, where its value changes ;)
* Example: State-full object with nested objects hierarchy, like in games development.

## Snapshot Debugging -> Time Travel Debugging

## Live Unit Testing

This runs every time your code changes! As you keep typing your test keeps running in the background. Test results update live in the editor without leaving your cursor position.

## Unicode Character Recognition

👽 Yes! you don’t have to switch to Microsoft word for the character recognition!

# Refactoring – 3 Keys!

## Control + Dot – Context Refactoring

You can get assisted refactoring all from the light bulb! In the document, based on cursor.

## Control + T – Navigation shortcuts

## Control + Q – VS Menu/VS Tooling Options

## Loop family Refactoring

* Foreach to LINQ
* LINQ to Foreach
* For to Foreach
* Foreach to For

## Code Cleanup

Setup rules to follow for code clean up. This lives in .editorconfig file and can be shared across team members.

# Azure Tools

Built in templates for Azure types

* Azure functions
* ASP.NET Core in Azure
* Emulators inside of Visual Studio like cosmos db, storage etc.to develop locally
* Cloud disconnected mode
* Offline to Online -> Publish -> Azure Targets -> Azure App Service etc.
* Add sql storage more services in Visual Studio at a later point in publish profile from initial setup
* Choose existing Azure resources, when you are not authorized to create RG in azure