|  |  |  |  |
| --- | --- | --- | --- |
| **Topic** | **C#** | **Python** | **Comments** |
| Quotes | "the quick brown fox" | 'the quick brown fox' | In python, quote and apostrophe mean the same thing but people seem to prefer apostrophe |
| Quotes - preserve escape characters | @"preserve \n" | r'preserve \n' |  |
| Quotes - expand variables | $"name is {name}" | f'name is {name}' |  |
| Quotes - preserve embedded newlines | @"preserve  embedded newlines" | '''preserve  embedded newlines''' | In Python you can expand variables with f'''{name}''' syntax |

|  |  |  |  |
| --- | --- | --- | --- |
| **Topic** | **C#** | **Python** | **Comments** |
| Common Data Types | string | **str** |  |
|  | int | int | (same) |
|  | bool | bool | (same) |
| Common Collections | int[] |  | Don't use arrays in Python, use list instead |
|  | List<string> | **list [ ]** | Lists aren't typed |
|  | Dictionary<string, string> | **dict { }** | Dicts aren't typed |
|  | Tuple<string, string, string> | **tuple ( )** | Tuples aren't typed |

|  |  |  |  |
| --- | --- | --- | --- |
| **Topic** | **C#** | **Python** | **Comments** |
| Flow Control Keywords | if(loop < 100) { } | if loop < 100: | Don't need parantheses around if condition |
|  | else if(loop < 200) { } | elif loop < 200: |  |
|  | while(loop < 100) { } | while loop < 100: | Don't need parantheses around while condition |
|  | foreach(item in list) { } | for item in list: | Don't need parantheses around for condition, don't need to explicitly use KeyValuePair |
| Error Handling | try/catch/finally | try/**except**/finally | Note "except" instead of "catch" |

|  |  |  |  |
| --- | --- | --- | --- |
| **Topic** | **C#** | **Python** | **Comments** |
| Operators | && | and | Logical and |
|  | || | or | Logical or |
|  | ! | not | Logical not |
|  | i++ |  | **not supported in Python** |
|  | i += 1 | i += 1 | (same) |
| Boolean keywords | true | **True** | Capitalization matters |
|  | false | **False** | Capitalization matters |
| Null keyword | null  if(a == null)  {     result = "undefined";  } | **None**  if a is None:      result = 'undefined' | Capitalization matters |