# Scratch to C# : https://lh3.googleusercontent.com/BAoqJn0OYJYxyg4VmRwVsqJhOMR59mLEuNs2p4YBUz2MhQFYaPcv4alkRC00bSsPSBvr39QXiJkTS-AofKV5uy-RZnCxg1nj2KPV52UllXBZWzvqZhWV1i_M_cl7pwVm

|  |  |  |
| --- | --- | --- |
| https://lh4.googleusercontent.com/YXP82FEpR8gEUr9Jpn7TQbK2vxk_0moovIoW8u_SuCFdYYpOq4ymp-srFGWaHSB3GtUkuQY9y-3cxKdtTZ-CkP-y9BVYOoKue-U-_kvGOdvC5w4cHc1XnCsdiEfB07X3 | + | Add two values |
| https://lh4.googleusercontent.com/Q39fMySmXJwkh3PN49TS29N-TsbV8hw1DIG1bDgTl0e28H4UGC-TbgSC2Hf3X6Q0e4N8VCZbAeKoso-_6h1QXIqM3RSwY3zEGXiaWdUB_kMKjNyUOcO6tad_6Rf8Ghhu | - | Subtract 2nd value from first |
| https://lh3.googleusercontent.com/e_tAc8bWhh8nQXpx3FnOd6jk3cChOT-zzBXnlWxYxVXRcTpwcAN5NG_UiGhR-rm5GWAt5U8oRqru2EHrbzDrwPejQmsTwsmNYS3l6zNi87zTejAt1co8uAQBfPLYUfO6 | \* | Multiply 2 values |
| https://lh5.googleusercontent.com/ZRiMNLwfBzYd8JkvcQAjNsLqcjj5BQQ74b0m8lWtCCTBMVDYdR4gF5wmEqlYXedToNtYGZzr763wO6LpRcdFkLxa4ObU0AS1tbpvLgwyzGSzLm6eKLNdd5HSYoIOm5hg | / | Divide first value by 2nd value |
| https://lh4.googleusercontent.com/Kiv--NJC7A5cD7Q4qRTOTnqX5n98xwGsX8xyW7biiRx-fIOnMCQBPE87O6WTV6DBCpCIKkbgFkfGxO9K7-ct35BOLfs3Im4bc88j7A4_OXL59UZ8ImDanoOmeUtk7iD7 | Random.Next(1,10) | Pick a random number in the specified range. |
| https://lh3.googleusercontent.com/bObA6T2q1cXnGH-71heQkdJDiTPayuROG6KRiv9gr1neScRnIg8AaCijp_XKBgf3kAkyOomlHPZuYmFIkEx4x9nF8O352YTCTGdxCTCnlHvVJz896LJO0tRGRigjttsc | < | Is the left less than the right |
| https://lh3.googleusercontent.com/CajFRJYIuWMbDEN8mJA7iBr9Gn9ceS5D-w1hHB8Ea1a4g5XusYLxrJ7GbhX0FDxcWnTfHcF9aJvjDaLZ6rUc-7T2E5r1mn1QE8RQQa5T9tfVxMUqqaT0JFKWITBJ_osx | == | Are two values equal |
| https://lh4.googleusercontent.com/EsmZmyPeoKdOQrAU7ykiIXWyv5cDNMNgcc-_eUA7lX3IC62fDnMq2bdL0jwTHIeeBnv4E6_oCfP5blbnwT-CSRBqPMyxglIL1tTYVDSvKS0m9Uf66mDjfIrIGOzRL7Ky | > | Is the left more than the right |
| https://lh4.googleusercontent.com/fpAM9GtlasGzWjBbNRZjZOLFSt34ZhGzJeEuKb8vvXEWRlT_5ZHGOtKmthRrBWkwOUI_hV7nq76dQAXCA1uZQB3JDok9bifs_jR4v_WSfYYPI8zTdGA5b_vwHhDwWX8B | && | Are both conditions true |
| https://lh4.googleusercontent.com/6ZkzxjbAMwFYQ0KRQx0D4zfu2sPFYwfY8GbwQ-xDdXlk-g4CVLggsES6LacxYphofi7kQe4MKcgwnWiHkuTO4MBhNruEc9sryHc0zQZW3mnvW2nzu4dqlotAe2AEUnts | || | Is either condition true |
| https://lh4.googleusercontent.com/zribUAI57O5Rmw41Z-eBrcTUmpZmbBCOwDRuo8ZkL83YkMdKLNjRsOFPib4rCtTTP1HoJ-ZZY7LZNfG_m8sIpvATq6ohMgmnEV0qYtPVzTWZ4eNfNEHtnnUjN5SAocs0 | ! | convert true to false and vice versa |
| https://lh5.googleusercontent.com/rkT3e3q3FNk27iBx-yPtgnsJaKSK1XTm973EYK8kxrdVwk3FF2sevx3RMySE42L0CIJdmG1WAO9M1scUkTV2UF80ttKAWH1Jk4ywkhJfv7c1W3vu94zn5m50_a_Sne2A | "hello " + "world" | combine two strings. |
| https://lh6.googleusercontent.com/dV9uZ7KfC-9xwa4iRJj9rYTwIwWl7kQdo85sLtKV5L20fvalJN5dJf2X0uatbWe_iLqIvHXFlI7i4fqMC5S2qNMYmJnHEwI8WCbqSTA7-zAMoUUO-ANXg8hurf6tjsdX | "world"[0] | Get first character in a string. Array indexes are zero based in C#, not 1 based like in Scratch. |
| https://lh3.googleusercontent.com/-n3ZowaCbmVxDAxT7wYAD6hf88HU_Y3IuPc9cQYlwkyMFhLa9AkR0U-2v4mLMLzoHIQXBo2Qk6u_A6LwcMi06-ohNc_TnB_ziEvr-qHgb9wowTSO3_W5wGYMDNJcng4h | "world".length() | Get the length of a string |
| https://lh3.googleusercontent.com/Z6AfOdMswHZ1alf_02MZpct1fDVB0czrqHlGNmwnoK5pVbwd2ysj-94fQlQYt_u2HmsddAwGM3jdt_StV2KB5jxLJSTyt4QS07Vjvxy-k-J56wHI5eFOGthrNuZOc1IJ | % | Get the remainder after dividing the first number by the 2nd.  For example:  5 % 2 = 1  Using %2 you can determine if the first number is even or odd.  In this case, the result is 1, so therefore 5 is odd.  You can also determine if a number divides evenly by 3  9 % 3 = 0 |
| https://lh5.googleusercontent.com/HrbFDN7Vm7MPFeK3ZWR1UDQ8VS_KdcjlRtPql_MtzScJNna5sAHSSKDtqOfccjuWMwmpwAkNfPmw9IRyPD-30WsEZ14x4jTVHHdc0T5Bf06CKVl9HWs1C0LMWAaU8kzq | Math.Round(MyValue, 0) | Round the value to 0 decimal places. In C# we can specify the number of decimal places. |
| https://lh5.googleusercontent.com/vg8b2oIx3WKts3LfBKCi3_q9OZz8PlBhDXqgvEN_QuF0dooBEYBHQy1P0qGruTu7vUZQlmwKmCWeCDaHby2KwP6d3gzPZsDiGt9f5lpnR3RX0Df0CWSzocLIXlowj0OP | Math.Abs(10) | Returns the absolute value of a number.  This means if the number is more than zero, then it doesn’t change.  But if it’s less than zero, then it becomes the same number above zero.  Math.abs(5) = 5 Math.abs(-5) = 5 |
| https://lh3.googleusercontent.com/3Ogwy4rBt0dj-QGLmK781mvVw2Ht_SUEuriVZp0HJsHzY6FKp7COB47wC_VafNwJVpT9TJh5uNbsJPWo4GICXNeBvfMToStirOi8vpalJDRkud5_lQCi5oGHzVTQfMsY | Math.Sqrt(10) | Returns the square root of a number.  This is the number that would have to be multiplied by itself to get the number requested. Math.Sqrt(9) = 3 |
| https://lh6.googleusercontent.com/9PwTV48s1uy3fMdAX2r2Pkpf9dOMNd6b5RuSIjgmOZtvXN8EZcff7FqwqkUpxp4JOir7is5-KHwDWUAiwFzrXLCpCSALP277-YU0tB21NPzYX4Che0vQdtrK7Wh4HNtH  https://lh4.googleusercontent.com/z3UM0Ibe4jA4_WnMXnby8pfoK6uibQuka_cjrFbEhWgxPsYMDsMlUMiJTmGEKrcx2MR2KH48qWAOlXTiN5Q6buDJZS8BBJvzCWrvKKankbfLKy-_nQtchrTOWE528Aun | Math.Sin(0.17)  Math.Cos(0.17) | Returns the sine or cosine of a an angle in radians.  To use degrees instead of radians, you’d do it like this  Math.Sin(MathHelper.ToRadians(10))  These are helpful for circular movements by setting X to Sin(time) \* Radius and Y to Cos(time) \* Radius |
| https://lh5.googleusercontent.com/QXxP9tP4y7Mv9Oj7b5sAnlcZwjqI_V8l_x60d8Gy3v0etfqnAloNdjnu0HmDxiYnWYlyh5_zhwXtk5A49eeKunvMrCMXwA0OdlDapKv8X360_AVEt77eKt1tS6p4-WVF | Math.Tan(0.17) | Returns the tangent of the specified angle (in radians). This is the ratio of the opposite side to the adjacent side.  It’s the slope of the angle. |
| https://lh6.googleusercontent.com/BqNb9gXZz_wM31Pq6Jjfe1IGbyG77D8T44x1bFY7IPsra4_DU-5b9CjGqG9wuNp4NUU-txxBPf1VR2Rd5fW1birSorlYArK6DvhH94kFvd_xtnFhdIndBnlHZ3yzvS_p  https://lh5.googleusercontent.com/IIdhFAIvXHF_Xfoa2DAW9ZDmDOyVKTb6x7W9SJNq0Nki9mEo7rPJgt3CdT4fG7uSS7Yq-MtdPSQkUiVeX0edSrUDtBMQQZN94HPJhSyG-XLMNrcuXY6osBz95JNLGdv2  https://lh6.googleusercontent.com/cKIo7BiX-_--v7qT8xX_CA3ewkqO8NKL7C3qcZXaXtKMmGwrFzABx2HPbwr3_bBta7PydWzHcExZbYxogD9qxou-ux7L8rplkzTw3XpuIKmxyeenaCWC-0bppT3xUou6 | Math.Asin(0.5)  Math.Acos(0.5)  Math.Atan(0.5) | Inverse trigonometric functions. |
| https://lh5.googleusercontent.com/vLSkVasfewmCCNN6HgxpjqPsN-I2Qm9DI_Vs69p1sIdHtE7XjT4NWqF4O5kxdKjuPmm7x8jd_sykFVADo56p4vU_PBC0TQ6JE5Ol13TY9OySXaeHC5XknuIZ0F8_dr8b | Math.Log(10) | Returns the natural logarithm of a specified number. |
| https://lh5.googleusercontent.com/pS80v70oPSc8FiOTmc8js4jId5XtZ332KaW0QxuGV6xXun0jemjK25ozxnHYu18yltM7Ixp-WW0YASoJ9qWXz1AZucKA1jR32mrid6z5E-u595t_xsX6-CasnhrAsA_y | Math.Log10(10) | Calculates the base 10 logarithm of a number. |
| https://lh4.googleusercontent.com/JiYR97_1dIP6k5i36ZCn7Oh6uVR7tMUiSaaqXxtyLEkL0_qadLvXZOc7PlMBfxgZQLMhMkdEzYbmMfVpjtOL8CiHTqL5gVb5U8TD7K41TKqJSKXFYy2HU4QNRmh0EOK5 | Math.Exp(10.0); | Calculate the exponential value of a number |
| https://lh3.googleusercontent.com/aHWVssc3Mzc6Jd1Ip1SPtq67ORU2LigSm-nVEYe3dNgNqAFyh9CvhdsfcxjK32vXshlzp8ZWpRXA812S3DL0fphgD2d8JnHp6T3pbgoCzO38iq4MuYtT3yyzX0GPNCxB | Math.Pow(10.10); | The power is the value of one number or expression raised to another number. |