console.log(10 + 10);

**Output: 20**

**explanation: both numbers are added**

console.log(10 + "10");

**Output:1010**

**String is converted into number by arithmetic operator**

**The + operator concatenates the two strings, resulting in "1010"**

console.log(10 + +"10");

**Output: 20**

console.log(10+"10"+10);

**Output:101010**

**String is converted into number by arithmetic operator**

**The + operator concatenates the strings, resulting in "101010"**

console.log(10+ +"10" + 10);

**Output:30**

**The unary plus operator (+) is used to convert the string "10" to a number**

**The expression now becomes console.log(10 + 10 + 10)**

console.log(10 - "2");

**Output:8**

**The unary operator (-) is used to convert the string to a number**

console.log(10 - "2" - "8");

**output:0**

**The unary operator (-) is used to convert the string to a number**

**Resulting in 10-2-8=0**

console.log(10+"2" - "2");

**Output:100**

**10+"2" results in "102" (string concatenation). Then, "102" - "2" converts both to numbers and performs subtraction: 102 - 2**

console.log(10>9>8);

**Output:False**

**10 > 9 evaluates to true.**

**true is converted to 1 when compared to 8.**

**1 > 8 evaluates to false.**

**console.log outputs false**

console.log(10 \* "10");

**Output:100**

**“10" is converted to the number 10 by \*.**

**10 \* 10 is evaluated to 100.**

**console.log outputs 100.**

console.log(100 / "100");

**Output:1**

**String 100 is converted to number so it becomes100/100=1**

console.log(100/"0");

output:infinity

string 0 is converted to number so 100/0 is infinity in js

console.log(100 + +"100" - "100" \* "100");

Output: -9800

**unary operator converts strings into numbers so it is 200- 10000**

console.log(1 == "1");

**Output:true**

**==operator converts the string "1" to the number 1. So 1=1 is true**

console.log(1 === "1");

**Output:False**

**=== operator does not perform type coercion, so it compares both the value and the type.**

**The number 1 and the string "1" have the same value but different types.**

console.log(1 == "one");

**Output: False**

**String one is not equal to number 1**

console.log(1 === "one");

**Output: False**

**String one is not equal to number 1**

**equality operator (===) checks both the value and the type of the operands.**

console.log(1+true);

**Output:2**

**true is converted to 1**

**1+1=2**

console.log(1 - true);

**Output:0**

**true is converted to 1**

console.log(1 + true - false);

**Output:2**

**false is converted to 0, true is converted to 1**

console.log("1" + true);

**Output: 1true**

**The + operator is used for string concatenation because one of the operands is a string.**

**"1" is a string and true is a boolean. When concatenating, true is converted to the string "true".**

console.log(+"1" + true);

**Output:2**

**The unary + operator before "1" converts the string "1" to the number 1.**

**true is implicitly converted to the number 1 in the arithmetic addition.**

console.log(undefined == undefined);

**Output: true**

**== operator checks for equality without considering the type. Since both operands are undefined, the result is true.**

console.log(undefined === undefined);

**Output:True**

**both operands are undefined and of the same type, the result is true.**

console.log(null == null);

**Output:True**

**Both are null so true.**

console.log(null === null);

**Output: false**

**both operands are null and of the same type, the result is true.**

console.log(undefined == null);

**Output:false**

**undefined and null are considered equal when using the == operator**

console.log(undefined === null);

**Output:False**

**undefined and null are of different types and different values.**

console.log(2+NaN);

**Output:Nan**

**Any arithmetic operation involving NaN will result in NaN**

console.log("2"+NaN);

**Output: 2NaN**

**"2" is a string and NaN is a special value representing "Not-a-Number".**

**The result is the concatenation of the string "2" and the string representation of NaN.**

console.log("2"+undefined);

**Output:2undefined**

**"2" is a string and undefined is a special value.**

**The result is the concatenation of the string "2" and the string representation of undefined.**

console.log(2+undefined);

**Output: NaN**

**The + operator is used for arithmetic addition, but undefined is not a number.**

**In this case, undefined is implicitly converted to NaN, and any arithmetic operation with NaN r**esults in NaN.

console.log(typeof "123");

**Output:123**

**The typeof operator returns the type of its operand as a string.**

**"123" is a string.**

console.log(typeof 2);

**Output: number**

**The typeof operator returns the type of its operand as a string.**

**2 is a number**.

console.log(typeof true);

console.log(typeof undefined);

console.log(typeof null);

console.log(typeof []);

console.log(typeof 1n);

console.log(typeof 1n+2n);

console.log(typeof 1+2n);

console.log(typeof 1/1n);