Concept of Binding:
The process of assigning values to attributes of program entities.
A set of attribute-value pairs of program entities.

Binding Time: the time a program entity’s attribute is assigned a value.
Language Design time
Language Implementation time
Compile time
Linking time
Load time
Run time

Attributes of Variables:
Name:
Address: l-value
Value: r-value
Lifetime: the period of time a storage area is allocated to a variable
Type: domain of values and a set of operators
Scope: the set of statements where a variable is visible, i.e., can be referenced.

Characterized variables in terms of storage binding and lifetime:
Static variables:
    History-sensitive:
Stack-Dynamic variables:
Explicit heap-dynamic variables:
Implicit heap-dynamic variables:

Characterized variables in terms of type bindings:

Type Checking:
Compatibility of Types:
    Name-equivalence:
    Structural-equivalence:

Characterized variables in terms of scope:
Concept of blocks:
Static scoped:
Dynamic scoped:

Referencing Environment: a set of names visible in a statement.

Named Constant: a variable that is bound to a value only one.
Static binding named constants (manifest constants): only constant values, named constants, and their combinations are allowed.
   Fortran 95:
   C#:  const

Dynamic binding: allowing expressions containing variables to be assigned to constants
   C++:  const
   C#:  readonly
   Java:  final

**Initialization**: the binding of a variable to a value at the time it is bound to storage.