everything is object in python and javascript
This statement is said little loosely said by people
They actually mean to say everything is instance
of the class(in python) or function(in javascript)
which has inherited from most basic object.

Here object are termed as instances of classes
in python and its documentation , but object is also used as term to identify most basic entity the building block of language i.e the most base class

Instances of classes are mentioned as objects everywhere in python docs but here we are referring to object as the class which is most basic , the firstever root class of python hence in these texts whenever we wish to refer instances the we will write instances only, we wont write objects there

Here is main diffirence in javascript and python :
-> javascript is functional purely so implements
inheritance using long prototype chains .
Every instance stores key value pairs of attributes:values
and methods:functionbody . And moreover each
object(i.e.instance) is function , and has attribute constructor
pointing to its parent function.There is no concept of class
inside the language and docs even tell today classes
in javascript are just syntactic sugar over constructor
functions.

->But whereas python is purely object-oriented language
like java. So here everything is instance of class which
inherits from the most basic base class called as object.It is most
basic building block of python. But here everything is public.
This is actual correct fixed implementation of inheritance
which javascript wants to implement but couldnt due to its
functional approach.Here in python classes and functions dont exist seprately and instead only classes and instances exist , even functions are instances.
But only thing is everything is public.Hence python follows purely only classes approach
In Python, functions are instances of the built-in function class which inherits from most basic object class

```
>>> help(object)
Help on class object in module builtins:

class object
 |  The base class of the class hierarchy.
 |
 |  When called, it accepts no arguments and returns a new featureless
 |  instance that has no instance attributes and cannot be given any.
 |
 |  Built-in subclasses:
 |      anext_awaitable
 |      async_generator
 |      async_generator_asend
 |      async_generator_athrow
 |      ... and 90 other subclasses
 |
 |  Methods defined here:
 |
 |  __delattr__(self, name, /)
 |      Implement delattr(self, name).
 |
 |  __dir__(self, /)
 |      Default dir() implementation.
 |
 |  __eq__(self, value, /)
 |      Return self==value.
 |
 |  __format__(self, format_spec, /)
 |      Default object formatter.
 |
```

```
|      Return str(self) if format_spec is empty. Raise TypeError otherwise.
|
|  __ge__(self, value, /)
|      Return self>=value.
|
|  __getattribute__(self, name, /)
|      Return getattr(self, name).
|
|  __getstate__(self, /)
|      Helper for pickle.
|
|  __gt__(self, value, /)
|      Return self>value.
|
|  __hash__(self, /)
|      Return hash(self).
|
|  __init__(self, /, *args, **kwargs)
|      Initialize self.  See help(type(self)) for accurate signature.
|
|  __le__(self, value, /)
|      Return self<=value.
|
|  __lt__(self, value, /)
|      Return self<value.
|
|  __ne__(self, value, /)
|      Return self!=value.
|
|  __reduce__(self, /)
|      Helper for pickle.
|
|  __reduce_ex__(self, protocol, /)
|      Helper for pickle.
|
|  __repr__(self, /)
|      Return repr(self).
|
|  __setattr__(self, name, value, /)
|      Implement setattr(self, name, value).
|
|  __sizeof__(self, /)
|      Size of object in memory, in bytes.
|
|  __str__(self, /)
|      Return str(self).
|
|  ----------------------------------------------------------------------
|  Class methods defined here:
|
|  __init_subclass__(...)
|      This method is called when a class is subclassed.
|
|      The default implementation does nothing. It may be
|      overridden to extend subclasses.
|
|  __subclasshook__(...)
|      Abstract classes can override this to customize issubclass().
|
|      This is invoked early on by abc.ABCMeta.__subclasscheck__().
|      It should return True, False or NotImplemented.  If it returns
|      NotImplemented, the normal algorithm is used.  Otherwise, it
|      overrides the normal algorithm (and the outcome is cached).
|
```

```
 |  ----------------------------------------------------------------------
 |  Static methods defined here:
 |
 |  __new__(*args, **kwargs)
 |      Create and return a new object.  See help(type) for accurate signature.
 |
 |  ----------------------------------------------------------------------
 |  Data and other attributes defined here:
 |
 |  __class__ = <class 'type'>
 |      type(object) -> the object's type
 |      type(name, bases, dict, **kwds) -> a new type
```

hence to see the chains of inheritance formed
by this class-based-inheritance we can use  inspect module

we can see ancestors of any class using getmro method

```
>>> inspect.getmro(int)
(<class 'int'>, <class 'object'>)
>>> inspect.getmro(object)
(<class 'object'>,)
>>> inspect.getmro(str)
(<class 'str'>, <class 'object'>)
>>>
```

we can see child classes of any class by __subclass__() method
on any class, see below we will use pretty print to see neatly
which classes have inherited the object class

```
>>> import pprint
>>> pprint.pp(object.__subclasses__())
[<class 'type'>,
 <class 'async_generator'>,
 <class 'bytearray_iterator'>,
 <class 'bytearray'>,
 <class 'bytes_iterator'>,
 <class 'bytes'>,
 <class 'builtin_function_or_method'>,
 <class 'callable_iterator'>,
 <class 'PyCapsule'>,
 <class 'cell'>,
 <class 'classmethod_descriptor'>,
 <class 'classmethod'>,
 <class 'code'>,
 <class 'complex'>,
 <class '_contextvars.Token'>,
 <class '_contextvars.ContextVar'>,
 <class '_contextvars.Context'>,
 <class 'coroutine'>,
 <class 'dict_items'>,
 <class 'dict_itemiterator'>,
 <class 'dict_keyiterator'>,
 <class 'dict_valueiterator'>,
 <class 'dict_keys'>,
```

<class 'mappingproxy'>,
<class 'dict_reverseitemiterator'>,
<class 'dict_reversekeyiterator'>,
<class 'dict_reversevalueiterator'>,
<class 'dict_values'>,
<class 'dict'>,
<class 'ellipsis'>,
<class 'enumerate'>,
<class 'filter'>,
<class 'float'>,
<class 'frame'>,
<class 'frozenset'>,
<class 'function'>,
<class 'generator'>,
<class 'getset_descriptor'>,
<class 'instancemethod'>,
<class 'list_iterator'>,
<class 'list_reverseiterator'>,
<class 'list'>,
<class 'longrange_iterator'>,
<class 'int'>,
<class 'map'>,
<class 'member_descriptor'>,
<class 'memoryview'>,
<class 'method_descriptor'>,
<class 'method'>,
<class 'moduledef'>,
<class 'module'>,
<class 'odict_iterator'>,
<class 'pickle.PickleBuffer'>,
<class 'property'>,
<class 'range_iterator'>,
<class 'range'>,
<class 'reversed'>,
<class 'symtable entry'>,
<class 'iterator'>,
<class 'set_iterator'>,
<class 'set'>,
<class 'slice'>,
<class 'staticmethod'>,
<class 'stderrprinter'>,
<class 'super'>,
<class 'traceback'>,
<class 'tuple_iterator'>,
<class 'tuple'>,
<class 'str_iterator'>,
<class 'str'>,
<class 'wrapper_descriptor'>,
<class 'zip'>,
<class 'types.GenericAlias'>,
<class 'anext_awaitable'>,
<class 'async_generator_asend'>,
<class 'async_generator_athrow'>,
<class 'async_generator_wrapped_value'>,
<class '_buffer_wrapper'>,
<class 'Token.MISSING'>,
<class 'coroutine_wrapper'>,
<class 'generic_alias_iterator'>,
<class 'items'>,
<class 'keys'>,
<class 'values'>,
<class 'hamt_array_node'>,
<class 'hamt_bitmap_node'>,
<class 'hamt_collision_node'>,

<class 'hamt'>,
<class 'sys.legacy_event_handler'>,
<class 'InterpreterID'>,
<class 'line_iterator'>,
<class 'managedbuffer'>,
<class 'memory_iterator'>,
<class 'method-wrapper'>,
<class 'types.SimpleNamespace'>,
<class 'NoneType'>,
<class 'NotImplementedType'>,
<class 'positions_iterator'>,
<class 'str_ascii_iterator'>,
<class 'types.UnionType'>,
<class 'weakref.CallableProxyType'>,
<class 'weakref.ProxyType'>,
<class 'weakref.ReferenceType'>,
<class 'typing.TypeAliasType'>,
<class 'typing.Generic'>,
<class 'typing.TypeVar'>,
<class 'typing.TypeVarTuple'>,
<class 'typing.ParamSpec'>,
<class 'typing.ParamSpecArgs'>,
<class 'typing.ParamSpecKwargs'>,
<class 'EncodingMap'>,
<class 'fieldnameiterator'>,
<class 'formatteriterator'>,
<class 'BaseException'>,
<class '_frozen_importlib._WeakValueDictionary'>,
<class '_frozen_importlib._BlockingOnManager'>,
<class '_frozen_importlib._ModuleLock'>,
<class '_frozen_importlib._DummyModuleLock'>,
<class '_frozen_importlib._ModuleLockManager'>,
<class '_frozen_importlib.ModuleSpec'>,
<class '_frozen_importlib.BuiltinImporter'>,
<class '_frozen_importlib.FrozenImporter'>,
<class '_frozen_importlib._ImportLockContext'>,
<class '_thread.lock'>,
<class '_thread.RLock'>,
<class '_thread._localdummy'>,
<class '_thread._local'>,
<class 'winreg.PyHKEY'>,
<class '_io.IncrementalNewlineDecoder'>,
<class '_io._BytesIOBuffer'>,
<class '_io._IOBase'>,
<class 'nt.ScandirIterator'>,
<class 'nt.DirEntry'>,
<class '_frozen_importlib_external.WindowsRegistryFinder'>,
<class '_frozen_importlib_external._LoaderBasics'>,
<class '_frozen_importlib_external.FileLoader'>,
<class '_frozen_importlib_external._NamespacePath'>,
<class '_frozen_importlib_external.NamespaceLoader'>,
<class '_frozen_importlib_external.PathFinder'>,
<class '_frozen_importlib_external.FileFinder'>,
<class 'codecs.Codec'>,
<class 'codecs.IncrementalEncoder'>,
<class 'codecs.IncrementalDecoder'>,
<class 'codecs.StreamReaderWriter'>,
<class 'codecs.StreamRecoder'>,
<class '_abc._abc_data'>,
<class 'abc.ABC'>,
<class 'collections.abc.Hashable'>,
<class 'collections.abc.Awaitable'>,
<class 'collections.abc.AsyncIterable'>,
<class 'collections.abc.Iterable'>,

<class 'collections.abc.Sized'>,
<class 'collections.abc.Container'>,
<class 'collections.abc.Buffer'>,
<class 'collections.abc.Callable'>,
<class '_winapi.Overlapped'>,
<class 'os._wrap_close'>,
<class 'os._AddedDllDirectory'>,
<class '_sitebuiltins.Quitter'>,
<class '_sitebuiltins._Printer'>,
<class '_sitebuiltins._Helper'>,
<class 'types.DynamicClassAttribute'>,
<class 'types._GeneratorWrapper'>,
<class 'operator.attrgetter'>,
<class 'operator.itemgetter'>,
<class 'operator.methodcaller'>,
<class 'itertools.accumulate'>,
<class 'itertools.batched'>,
<class 'itertools.chain'>,
<class 'itertools.combinations'>,
<class 'itertools.compress'>,
<class 'itertools.count'>,
<class 'itertools.combinations_with_replacement'>,
<class 'itertools.cycle'>,
<class 'itertools.dropwhile'>,
<class 'itertools.filterfalse'>,
<class 'itertools.groupby'>,
<class 'itertools._grouper'>,
<class 'itertools.islice'>,
<class 'itertools.pairwise'>,
<class 'itertools.permutations'>,
<class 'itertools.product'>,
<class 'itertools.repeat'>,
<class 'itertools.starmap'>,
<class 'itertools.takewhile'>,
<class 'itertools._tee'>,
<class 'itertools._tee_dataobject'>,
<class 'itertools.zip_longest'>,
<class 'reprlib.Repr'>,
<class 'collections.deque'>,
<class 'collections._deque_iterator'>,
<class 'collections._deque_reverse_iterator'>,
<class 'collections._tuplegetter'>,
<class 'collections._Link'>,
<class 'functools.partial'>,
<class 'functools._lru_cache_wrapper'>,
<class 'functools.KeyWrapper'>,
<class 'functools._lru_list_elem'>,
<class 'functools.partialmethod'>,
<class 'functools.singledispatchmethod'>,
<class 'functools.cached_property'>,
<class 'enum.nonmember'>,
<class 'enum.member'>,
<class 'enum._not_given'>,
<class 'enum._auto_null'>,
<class 'enum.auto'>,
<class 'enum._proto_member'>,
<enum 'Enum'>,
<class 'enum.verify'>,
<class 're.Pattern'>,
<class 're.Match'>,
<class '_sre.SRE_Scanner'>,
<class '_sre.SRE_Template'>,
<class 're._parser.State'>,
<class 're._parser.SubPattern'>,

```
<class 're._parser.Tokenizer'>,
<class 're.Scanner'>,
<class 'ast.AST'>,
<class 'contextlib.ContextDecorator'>,
<class 'contextlib.AsyncContextDecorator'>,
<class 'contextlib._GeneratorContextManagerBase'>,
<class 'contextlib._BaseExitStack'>,
<class 'ast.NodeVisitor'>,
<class 'dis._Unknown'>,
<class 'dis.Bytecode'>,
<class 'warnings.WarningMessage'>,
<class 'warnings.catch_warnings'>,
<class '_tokenize.TokenizerIter'>,
<class 'tokenize.Untokenizer'>,
<class '_weakrefset._IterationGuard'>,
<class '_weakrefset.WeakSet'>,
<class 'weakref.finalize._Info'>,
<class 'weakref.finalize'>,
<class 'inspect.BlockFinder'>,
<class 'inspect._void'>,
<class 'inspect._empty'>,
<class 'inspect.Parameter'>,
<class 'inspect.BoundArguments'>,
<class 'inspect.Signature'>,
<class 'rlcompleter.Completer'>,
<class '__future__._Feature'>,
<class 'importlib._abc.Loader'>,
<class 'threading._RLock'>,
<class 'threading.Condition'>,
<class 'threading.Semaphore'>,
<class 'threading.Event'>,
<class 'threading.Barrier'>,
<class 'threading.Thread'>,
<class 'importlib.util._incompatible_extension_module_restrictions'>,
<class 'typing._Final'>,
<class 'typing._NotIterable'>,
typing.Any,
<class 'typing._PickleUsingNameMixin'>,
<class 'typing._TypingEllipsis'>,
<class 'typing.Annotated'>,
<class 'typing.NamedTuple'>,
<class 'typing.TypedDict'>,
<class 'typing.NewType'>,
<class 'typing.io'>,
<class 'typing.re'>,
<class 'platform._Processor'>,
<class 'ipaddress._IPAddressBase'>,
<class 'ipaddress._BaseConstants'>,
<class 'ipaddress._BaseV4'>,
<class 'ipaddress._IPv4Constants'>,
<class 'ipaddress._BaseV6'>,
<class 'ipaddress._IPv6Constants'>,
<class 'urllib.parse._ResultMixinStr'>,
<class 'urllib.parse._ResultMixinBytes'>,
<class 'urllib.parse._NetlocResultMixinBase'>,
<class 'textwrap.TextWrapper'>,
<class 'traceback._Sentinel'>,
<class 'traceback.FrameSummary'>,
<class 'traceback._ExceptionPrintContext'>,
<class 'traceback.TracebackException'>,
<class 'pydoc.Doc'>,
<class 'pydoc.Helper'>,
<class 'pydoc.ModuleScanner'>,
<class 'zlib.Compress'>,
```

```
 <class 'zlib.Decompress'>,
 <class 'zlib._ZlibDecompressor'>,
 <class '_bz2.BZ2Compressor'>,
 <class '_bz2.BZ2Decompressor'>,
 <class '_lzma.LZMACompressor'>,
 <class '_lzma.LZMADecompressor'>,
 <class '_random.Random'>,
 <class '_sha2.SHA224Type'>,
 <class '_sha2.SHA256Type'>,
 <class '_sha2.SHA384Type'>,
 <class '_sha2.SHA512Type'>,
 <class 'tempfile._RandomNameSequence'>,
 <class 'tempfile._TemporaryFileCloser'>,
 <class 'tempfile._TemporaryFileWrapper'>,
 <class 'tempfile.TemporaryDirectory'>,
 <class 'string.Template'>,
 <class 'string.Formatter'>,
 <class 'pydoc.TextDoc.docclass.<locals>.HorizontalRule'>,
 <class 'pydoc.TextDoc.docclass.<locals>.HorizontalRule'>,
 <class 'dataclasses._HAS_DEFAULT_FACTORY_CLASS'>,
 <class 'dataclasses._MISSING_TYPE'>,
 <class 'dataclasses._KW_ONLY_TYPE'>,
 <class 'dataclasses._FIELD_BASE'>,
 <class 'dataclasses.InitVar'>,
 <class 'dataclasses.Field'>,
 <class 'dataclasses._DataclassParams'>,
 <class 'pprint._safe_key'>,
 <class 'pprint.PrettyPrinter'>]
>>>

almost everything in python has inherited from base class called as object
Thus they all have methods and attributes of object class, and they override
and polymorph them whenever they wish
```

note integers , float , individual strings , booleans , None  all of them are instances of some classes which are visible by help( value ) , inspect.getclasstree() works for only iterable things in python