After pipes we have figured that one can built sockets on top of pipes where sockets are bidirectional.
Like we have socketpair(2), which creates an unnamed pair of connected sockets in the specified domain domain, of the specified type, and using the optionally specified protocol.

We need to create_socket, then attach this socket to a host, (i.e., binding), and then listen (in case if it is going to act as a server)

You cannot use any port because only limited ports are allowed for a system being open for the user applications.
Functions like getsockname, gethostname in various languages can give you back the information for a particular socket

Questions: Does socket act as a blocking or non-blocking?

Socket is just a file descriptor like any other file descriptor

connections are asymmetrical: one process requests a connection, the other process accepts the request

In socket programming:
We user accept(2), select(2) functions to listen and then receive the incoming stream. The concept remain same in every other language.

This whole process of socket connections could be counted in I/O Multiplexing

A snippet to listen/read the data:
while ((n = read(fd1, buf, BUFFSIZE)) > 0) { if (write(fd2, buf, n) != n) { fprintf(stderr, "write error\n"); exit(1); } }

Answer to previous question:-
You can either use blocking mode,or non-blocking mode. You can also use forking a child or using asynchronous I/O fd