In which way data structures support object-oriented development

Data Structures are software mechanisms for representing and manipulating data. They are just human-made constructs to tell a computer what to do step by step. Moreover, the object mechanism is a beneficial software mechanism. *Objects* are a data structure that contains abstract data in the form of attributes and behaviour to represent real-world objects. As a result, an object operates on its "own" data structure. On the other hand, the data structure has no behavioural structure because it only stores data and focuses on procedural programming.

Nevertheless, both data structures and objects can support each other, as they can cover each other's immunities and vulnerabilities. At the high level of software development, programmers must mix objects and functions for the best software quality.

Examples of how data structures can support object-oriented programming are the library classes like Map and List, which represent data structures. These classes are used to implement and set up a data structure that programmers can efficiently work within their programs by creating instances of them (i.e. objects).

References:

OpenDSA (2016) 4.2. Introduction to Object Oriented Programming. Available from: https://opendsa-server.cs.vt.edu/ODSA/Books/Everything/html/IntroOO.html [Accessed 18 July 2022].

Uncle Bob Consulting LLC (2007) Active Record vs Objects. Available from: https://sites.google.com/site/unclebobconsultingllc/active-record-vs-objects#TOC-The-Difference-between-Objects-and-Data-Structures [Accessed 18 July 2022].