DBMS LANGUAGES

• Data Definition Language (DDL)
• Data Manipulation Language (DML)
• High-Level or Non-procedural Languages: These include the relational language SQL
• May be used in a standalone way or may be embedded in a programming language

• Low Level or Procedural Languages:

These must be embedded in a programming language

Data Definition Language (DDL)

Used by the DBA and database designers to specify the conceptual schema of a database.

• In many DBMSs, the DDL is also used to define internal and external schemas (views).
• In some DBMSs, separate storage definition language (SDL) and view definition language (VDL) are used to define internal and external schemas.
• SDL is typically realized via DBMS commands provided to the DBA and database designers

**Data Manipulation Language (DML)**

Used to specify database retrievals and updates DML commands (data sublanguage) can be *embedded in a general-purpose programming* language (host language), such as COBOL, C, C++, or Java.

• A library of functions can also be provided to access the DBMS from a programming language
• Alternatively, stand-alone DML commands can be applied directly (called a *query language*).

**Types of DML**

• **High Level or Non-procedural Language:**

  For example, the SQL relational language are “set”-oriented and specify what data to retrieve rather than how to retrieve it.
  Also called **declarative languages**.

• **Low Level or Procedural Language:**
  Retrieve data one record-at-a-time;
  Constructs such as looping are needed to retrieve multiple records, along with positioning pointers.

**DBMS Interfaces**

• Stand-alone query language interfaces
  Example: Entering SQL queries at the DBMS interactive SQL interface (e.g. SQL*Plus in ORACLE)
• Programmer interfaces for embedding DML in programming languages
• User-friendly interfaces
• Menu-based, forms-based, graphics-based, etc.

DBMS Programming Language Interfaces

• Programmer interfaces for embedding DML in a programming languages:
• **Embedded Approach:** e.g. **embedded SQL** for C, C++, etc., SQLJ (for Java)
• **Procedure Call Approach:** e.g. **JDBC for Java**, ODBC for other programming languages
• **Database Programming Language Approach:**
  e.g. ORACLE has PL/SQL, a programming language based on SQL; language incorporates SQL and its data types as integral components/

User-Friendly DBMS Interfaces

• Menu-based, popular for browsing on the web
• Forms-based, designed for naïve users
• Graphics-based (Point and Click, Drag and Drop, etc.)
• Natural language: requests in written English
• Combinations of the above: For example, both menus and forms used extensively in Web database interfaces

Other DBMS Interfaces

• Speech as Input and Output
• Web Browser as an interface
• Parametric interfaces, e.g., bank tellers using function keys.
• Interfaces for the DBA:
• Creating user accounts, granting authorizations
• Setting system parameters
• Changing schemas or access paths