Glossary

Date and Time Handling in Java

Welcome! This alphabetized glossary contains many terms used in this module. Understanding these terms is essential when working in the industry, participating in user groups, and participating in other certificate programs.

Term	Definition
database storage	A method of storing dates in a specific format required by databases.
date format	A pattern used to represent dates as strings, such as dd/MM/yyyy or yyyy-MM-dd.
date manipulation	The process of changing dates by adding or subtracting days, months, or years.
date validation	The process of checking whether a date is valid according to a specific format.
DateTimeFormatter	A Java class used to define patterns for parsing and formatting dates and times.
DateTimeParseException	An exception that is thrown when a date or time string cannot be parsed into a valid object, specific to Java's java.time.format package.
daylight saving time (DST)	A practice of moving clocks forward during warmer months to extend evening daylight.
defining format	The process of creating a DateTimeFormatter with a desired pattern.
delimiters	Characters or sequences used to separate strings into parts, such as commas or spaces.
Duration	A class representing a time-based amount of time, such as hours or minutes.
Epoch	The starting point of time used in computing, typically January 1, 1970.
epoch time	The number of seconds or milliseconds since January 1, 1970 (Unix epoch).
error message	A notification displayed when an operation, such as parsing, fails.
extraction	The process of retrieving specific information, such as dates, from a larger text.
Format Style	The predefined styles in DateTimeFormatter for date and time formatting, such as SHORT, MEDIUM, LONG, and FULL.
formatting	The act of converting a date or time object into a readable string format.
full weekday name	A representation of the full name of the day, such as Monday, Tuesday, etc.
get current date	The use of LocalDate.now() to get the current date.
import	A statement used in Java to include external classes or packages required for a program to run.
Instant	A class representing a moment on the timeline in UTC.
ISO-8601	A standard for representing date and time in an international format.
Java 8 Time API	A set of classes introduced in Java 8 under the java.time package for date and time manipulation.
LocalDate	A Java class that represents a date without time-zone information.
LocalDate.now()	A method that retrieves the current date from the system clock.
LocalDate.of()	A method used to create a LocalDate object with a specific year, month, and day.
LocalDate.parse()	A method in Java used to convert a string into a LocalDate object using a specific pattern.
LocalDateTime	A Java class representing both date and time without time-zone information.
LocalTime	A class representing a time that includes an hour, minute, and second without a date.
MonthDay	A class that represents a combination of a month and day without a year.
OffsetDateTime	A date-time with an offset from UTC/Greenwich.
UTC	A coordinated universal time standard used globally.
Year	A class that represents a year, such as 2025.
YearMonth	A class that represents a combination of a year and month without a day.
ZonedDateTime	A class representing a date and time with a time zone.
ZoneId	An identifier for a time zone, such as "Asia/Kolkata."
ZoneOffset	The amount of time added or subtracted from UTC to get local time.

Author