Part: Short Questions

Q1: What is Supabase?

Ans: Supabase is an "open-source alternative to Firebase" that provides a backend (database, authentication, real-time subscriptions) for applications without needing to manage a server.

Q2: Which database engine is Supabase built on top of?

Ans: It is built on top of **PostgreSQL**, which is one of the world's most powerful relational database systems.

Q3: State one major benefit of using Supabase.

Ans: You get the power of SQL (Structured Query Language) combined with an easy-to-use dashboard that looks and feels like a spreadsheet.

Q4: Differentiate between a Spreadsheet (Excel) and a Relational Database.

Ans: unlike Excel, where you can type anything anywhere, a Relational Database requires structure. Data is stored in strict **Tables**, and every **Column** must have a specific type (Text, Number, Date).

Q5: Why is the "Database Password" step considered crucial during setup?

Ans: It is crucial because you must create a strong password and save it immediately; Supabase will not show the password to you again after creation.

Q6: Why should you select a "Region" closest to you when creating a project?

Ans: Selecting a region closest to you (e.g., Singapore or Mumbai) ensures faster speed and lower latency for your application.

Q7: What is the standard convention for naming tables in Supabase?

Ans: The standard convention is to use **lowercase** and **plural** names, such as "sensors" or "students".

Q8: What is the purpose of the 'id' column?

Ans: The 'id' column is the **Primary Key**. It acts as the unique fingerprint for every row in the database and should not be removed.

Q9: Define "Schema" in a database context.

Ans: A Schema is the definition that tells the database what kind of data to expect. It defines the columns and their specific data types.

Q10: Which data type is best suited for storing temperature values?

Ans: The **float8** data type is used for decimal numbers, making it suitable for temperature readings (e.g., 25.5).

Q11: What is the purpose of the now() function in default values?

Ans: Setting a default value to now() allows the database to automatically record the exact time a piece of data is entered without manual input.

Q12: What does the timestamptz data type store?

Ans: It stores **Time data**, specifically timestamps that often include timezone information.

Q13: What are the two ways to insert data into Supabase?

Ans: Data can be inserted manually via the **GUI (Table Editor)** or via code using the **SQL Editor**.

Q14: Write the SQL command used to insert data into a table named 'sensors'.

Ans: INSERT INTO sensors (location, temperature) VALUES ('Kitchen', 28.0);

Q15: What is the 'Project URL'?

Ans: The Project URL is the specific web address assigned to your database, used to connect your IoT device or website to the cloud.

Q16: Name the two types of API Keys found in settings.

Ans: The two keys are the **anon** (Public) key and the **service_role** (Secret) key.

Q17: Which API key is safe to use in frontend website code?

Ans: The **anon (Public)** key is safe to use as it follows security rules (Row Level Security).

Q18: Why must you NEVER share the service_role key?

Ans: The service_role key bypasses all security rules and creates a "god mode" connection. If shared, it allows anyone to control or delete your entire database.

Q19: How do you access the visual database manager in Supabase?

Ans: You access it by clicking the **"Table" icon** on the left sidebar of the dashboard.

Q20: What is the first practical step to initialize the cloud environment?

Ans: Go to Supabase.com and sign in (usually using GitHub), then click "New Project".

Part: Comprehensive Questions

Q1: Explain the concept of Supabase and how it differs from a standard spreadsheet application.

Ans:

- **Supabase:** It is a backend-as-a-service built on PostgreSQL. It provides database management, authentication, and real-time features without server management.

- **Difference from Spreadsheet:** While spreadsheets (like Excel) allow unstructured data entry (typing anything anywhere), Supabase uses a Relational Database model. This requires strict structure where data is organized into **Tables**, and each **Column** enforces a specific data type (Text, Number, Bool) to ensure data integrity.

Q2: Describe the complete step-by-step process for creating a new project in Supabase.

Ans:

1. **Sign Up:** Go to Supabase.com and sign in (typically via GitHub).
2. **Start:** Click the green "New Project" button.
3. **Name:** Provide a clear name for the project (e.g., My-IoT-Class-DB).
4. **Password:** Create a strong Database Password. *Crucial:* Save this password immediately as it is shown only once.
5. **Region:** Select the server location closest to your physical location (e.g., Singapore) to optimize speed.

Q3: Elaborate on the "Table Editor" interface and the rules for creating a new table.

Ans:

- **Interface:** The Table Editor is the visual database manager accessed via the "Table" icon on the dashboard sidebar.
- **Creating a Table:** When clicking "Create a New Table," you must follow specific conventions:
    - **Naming:** Use lowercase and plural names (e.g., sensors not Sensor).
    - **Primary Key:** Ensure the automatically added id column remains. This serves as the unique fingerprint for every row, allowing the database to distinguish between records.

Q4: detailed note on "Designing a Schema" and the importance of Data Types.

Ans:

- **Schema:** The schema is the blueprint of the database. It tells the system exactly what data format to expect for each column.
- **Data Types:** You must assign specific types to columns:
    - **float8:** Used for decimal numbers (e.g., temperature).
    - **text:** Used for strings of characters (e.g., location).
    - **bool:** Used for True/False values (e.g., is_active).
    - **timestamptz:** Used for time data (e.g., created_at).

- **Default Values:** Functions like now() can be used to automatically populate fields like timestamps.

Q5: Compare and contrast the two methods of inserting data into Supabase (GUI vs. SQL). Ans:

- **Method 1: GUI (The Easy Way):** Performed in the Table Editor by clicking "Insert Row." A form appears where you type values (e.g., "Living Room", "25.5") and click Save. This is best for testing or small manual entries.
- **Method 2: SQL (The Code Way):** Performed in the SQL Editor. You write commands like INSERT INTO... VALUES... and click Run. This method represents how actual code and applications will communicate with the database programmatically.

Q6: Explain the significance of API Keys in Supabase and the security risks associated with them.

Ans:

- **Purpose:** API keys are required to connect external apps (IoT devices, websites) to the database. They are found in Settings > API.
- **Types:**
    1. **anon (Public):** Safe for frontend code. It respects Row Level Security (RLS) rules.
    2. **service_role (Secret):** Bypasses all security rules ("god mode").
- **Security Risk:** The service_role key must **NEVER** be shared or put in public code. If exposed, attackers can bypass security and delete or steal all data.

Q7: Write a note on connecting an IoT App to Supabase.

Ans: To establish a connection between an application (like an IoT sensor system) and the cloud database, two specific credentials are required:

1. **Project URL:** This is the specific web address endpoint for the database.
2. **API Key:** The application must send an API key (usually the anon key for public clients) to authenticate the request. These are located in the "Settings > API" section of the dashboard.

Q8: What is PostgreSQL and how does it relate to the Supabase platform?

Ans:

- **PostgreSQL:** It is one of the world's most powerful and advanced open-source relational database systems. It supports complex queries (SQL) and strict data structuring.
- **Relation to Supabase:** Supabase is the "Engine" built on top of PostgreSQL. Supabase takes the complexity of managing a raw PostgreSQL server and wraps it in an easy-to-use interface, making the power of SQL accessible to developers without deep server management knowledge.

Q9: Explain the structure and function of the SQL command used to insert data.

Ans: The command structure is: INSERT INTO table_name (column1, column2) VALUES (value1, value2);

- **INSERT INTO:** Tells the database we are adding new data.
- **table_name:** Specifies which table receives the data (e.g., sensors).
- **(columns):** Lists which specific fields we are filling.
- **VALUES:** Lists the actual data corresponding to the columns (e.g., 'Kitchen', 28.0).

Q10: Discuss the importance of the "Primary Key" in a Supabase table.

Ans:

- **Definition:** In Supabase, the Primary Key is usually the id column automatically added during table creation.
- **Function:** It provides a unique identifier for every single row in the table.
- **Importance:** Without a primary key, the database cannot distinguish between two rows that might have identical data (e.g., two sensors in the "Kitchen" reading "25.0"). It is the "unique fingerprint" required for efficient data management and retrieval.