Software Requirements Specification

For

Interior design database management system

By

202318061(Garvika Singh Rajawat)

and

202318005(Aditi Singh)

I. SRS

1. Case Study

1.1 Purpose

Interior designing is the art of decorating a residential home or commercial business according to the client's personal preference and style. It includes selecting color schemes, flooring materials, furniture artwork and accessories. It transforms an area into an aesthetically pleasing and budget friendly design.

Interior designing is simply outfitting a space with items you like.For some aesthetic of their home is so important to them that they enlist a professional interior designer to choose furniture, wall coverings,accessories, rugs and more.

The Software Requirements Specification's goal is to give a thorough and exhaustive breakdown of the functionalities, features, and specifications of the system. The development team's comprehension of what has to be built and the stakeholders' requirements are crucially connected by this document. The SRS establishes boundaries for the project by outlining the functionalities that will and won't be included in the system.

1.2 Intended Audience

The project's numerous stakeholders, each with a distinct level of technical skill and specialized interests, are expected to read the SRS document. The audience may include the client or product owner, the project manager, developers, and designers.

1.3 Product Scope

The product scope defines the boundaries that what and what not is included Product scope includes storing design ideas ,vendor’s information ,client management ,budget and expense tracking , different architect’s data and many more pertaining to the interior design of private property. Users will be able to -

1. Catalog furniture and supplies
2. Create quotations and invoices
3. Monitor design projects, and visualize design thoughts using the system .
4. The major product scope will be user authentication and authorisation , project management ,furniture and material catalog , design visualization , reporting and analytics, user collaboration.

1.4 Description

The IDDBMS provides a centralized platform for managing all aspects of interior design projects, including project planning, design concept development, material sourcing, budgeting, and progress tracking. It allows users to store and access project-related information, such as client requirements, design specifications, supplier details, and project timelines, in a secure and organized manner.

Definition and acronyms-

- SRS- Software requirement abbreviation
- CRM- Customer relationship management
- IDDBMS- Interior designing database management system

1. **Product Overview -**

Key features of the product:

- **Project Management:** The IDDBMS allows interior designers to create and manage projects, assign tasks, set milestones, and track progress. It provides a comprehensive view of project timelines, deliverables, and resource allocation.
- **Design Collaboration:** The system enables seamless collaboration between interior designers and clients. It allows designers to share design concepts, receive feedback, and iterate on designs in real-time. Clients can provide input, review designs, and approve changes through the system.
- **Supplier Management:** The IDDBMS facilitates the management of suppliers and sourcing of design materials. It provides a catalog of suppliers, allowing designers to search for specific products, compare prices, and place orders. Suppliers can also update inventory and delivery information through the system.
- **Financial Tracking:** The system includes features for budgeting and financial tracking. It allows designers to estimate project costs, track expenses, and

generate invoices for clients. Administrators can monitor financial data and generate reports for analysis.

- **Managing calendar:** Coordination with clients and other stakeholders is frequently necessary for interior designers. Designers can better organize their schedules and issue calendar invitations with reminders by using a digital calendar management system . Scheduling issues and last-minute adjustments can be reduced by confirming appointments the day before and putting cancellation procedures in place for unreliable clients.

2. User's interface -

IDDBMS system will support different user roles and permissions. Designers, administrators, and project managers may have different levels of access and functionality within the system. For example, designers may have the ability to create and modify design plans, while administrators may have access to system configuration and user management functionalities.

Each user interacts differently with the system. User interface varies for different user roles.

In IDDBMS the different users are -

i. Interior Designer

ii. Client

iii. Vendor

iv. Administrator

i. Interior Designer Interface:

The interior designer's interface allows them to easily manage projects, assign resources, and communicate with clients and vendors.

- **Home Page:** This is the landing page after login, showing an overview of ongoing projects, upcoming tasks, and notifications.
- **Project Management Page:** This page allows the designer to create, view, update, and delete projects. Each project can be expanded to show details like client information, project timeline, assigned resources, and progress.
- **Resource Management Page:** This page displays available resources that can be assigned to projects. It also allows the designer to order more resources from vendors.

- **Communication Page:** This page facilitates communication with clients and vendors. It might include messaging features and project update notifications.

ii. Client Interface:

The client interface enables clients to view project updates, approve designs, and make payments.

- **Home Page:** This page shows an overview of the client's projects, notifications, and upcoming appointments with the designer.
- **Project Details Page:** This page allows clients to view detailed progress reports for each project, including design drafts, timelines, and costs.
- **Payment Page:** This page allows clients to view invoices and make payments.

iii. Vendor Interface:

The vendor interface allows vendors to update inventory, process orders, and communicate with the designer.

- **Home Page:** This page shows an overview of orders, inventory status, and notifications.
- **Inventory Management Page:** This page allows vendors to update their inventory and view orders from designers.
- **Communication Page:** This page facilitates communication with designers.

iv. Admin Staff Interface:

The admin enables admin staff to manage user accounts, handle billing, and generate reports.

- **Home Page:** This page shows an overview of system usage, user accounts, and notifications.
- **User Management Page:** This page allows the admin staff to manage user accounts, including creating, updating, or deleting accounts.
- **Billing Management Page:** This page facilitates the handling of billing information and transactions.

- **Reports Page:** This page allows the generation of various reports, such as project status reports and resource utilization reports.

3. User Characteristics -

The IDDBMS is designed for interior designers and their team members, who may have varying levels of expertise and experience in the field. It is important for users of the system to possess basic computer literacy and familiarity with interior design processes and terminology. The system aims to accommodate different user roles, allowing designers, administrators, and project managers to interact with the system effectively.

4. Software interfaces:

In terms of software interfaces, the IDDBMS system will provide a user-friendly interface that is intuitive and easy to navigate. This can include features such as menus, buttons, and forms that allow users to interact with the system and perform tasks related to interior design, such as creating and managing projects, accessing design , and generating reports.

5. Hardware interfaces:

Regarding hardware interfaces, the IDDBMS system will require standard hardware components such as computers, servers, and storage devices. The system may also utilize input devices like keyboards as well as output devices such as monitors and printers.

The system may also need to integrate with other software systems or databases. This can include integration with external design software and project management tools.

6. Design and Implementation constraints

- **Budget:** The design and development of the IDDBMS will be constrained by the budget allocated for this project. This may limit the features that can be included in the initial release of the system.

- **Time:** The IDDBMS needs to be developed and deployed within a specific timeline. This may limit the amount of time available for testing and

refining the system.

- **Technology:** The IDDBMS may be constrained by the technology used for its development. For example, if a specific database management system or programming language is used, it will dictate the capabilities and limitations of the system.

- **Data Security and Privacy:** The IDDBMS will need to adhere to data security and privacy laws and regulations. This will constrain how data is collected, stored, processed, and shared within the system.

- **Business Rules:** The IDDBMS will be constrained by the business rules of the interior design industry. For example, there may be specific procedures for managing projects, interacting with clients, dealing with suppliers, and handling finances.

2. Requirement Collection

2.1 Background Reading

i. References and ii. Summary

https://wildcreativeproject.com/small-space/interior-design-process

The interior design process is then broken down into five major phases:

Client Briefing (Programming): This phase involves meeting the client and understanding their requirements.

Conceptualization (Schematic Design): In this phase, the designer begins conceptualizing the design and brainstorming solutions using the information obtained from the client.

The Creative Details (Design Development): This is where the designer fine-tunes the schematic design that the client approved earlier.

Preparing for Construction (Construction Documentation): At this point, the designer prepares the detailed working drawings and specifications..

https://www.geeksforgeeks.org/how-to-write-a-good-srs-for-your-project/

The link provided is an article titled "How to write a good SRS for your Project" on the GeeksforGeeks website. The article provides a detailed guide on how to write a Software Requirements Specification (SRS) for a software project.

https://www.studocu.com/in/document/visvesvaraya-technological-university/software-engineering/interior-report-159-dbms-based-mini-project/30558894

The link provided is a document titled "Interior report[159] - DBMS based mini project" from the Visvesvaraya Technological University. The document is a report on a mini project that utilizes a Database Management System (DBMS).

https://krazytech.com/projects/sample-software-requirements-specificationsrs-report-airline-database

The link provided is a Software Requirements Specification (SRS) report for an airline database system. The report outlines the system's functional and non-functional requirements, system features, and external interface requirements. The system stores flight details, customer descriptions, and reservation descriptions. It supports user privileges for customers and employees, with features for making reservations and managing flights. The system also includes non-functional requirements for data normalization and recovery methods in case of catastrophic failures.

https://yiren-wang.com/2019/12/15/database-design-and-implement-of-a-medium-sized-interior-design-firm/

The link provided is a blog post titled "Database Design and Implement of a Medium-sized Interior Design Firm" by Alan Yiren Wang. The blog post discusses the process of designing and implementing a database for a hypothetical interior design firm.

The blog post begins by outlining the problem: many small to medium-sized design companies, including the author's parents' company, lack a well-organized database that stores their data. This leads to issues such as difficulty finding historical blueprints and files, scattered product information from suppliers, unclear contributors to old projects, lost track of each stage of requirements from clients, and individual employees having their own collections of inspirational images and material maps with no channel to share

The blog post then describes the conceptual and logical design of the database. The design process follows three stages: conceptual, logical, and implementation. The conceptual level involves extracting business rules from interviews with stakeholders and potential users. The logical level involves creating an entity-relationship diagram (ERD) based on these rules. The implementation level involves using MySQL to implement the database on a local drive**.**

https://foyr.com/learn/interior-design-project-management-guide/

The link provided is a guide titled "Interior Design Project Management Guide" on the Foyr website. The guide offers a comprehensive overview of the project management process in interior design.

The guide begins by outlining the importance of project management in interior design. It emphasizes that project management is not just about managing time and resources, but also about managing the client's expectations and ensuring that the project is completed to their satisfaction.

The guide then delves into the different stages of interior design project management, including pre-design, design, construction, and post-project. Each stage has its own set of tasks and challenges. For example, in the pre-design stage, the designer needs to gather information about the client's needs and preferences. In the design stage, the designer needs to create a concept that meets the client's requirements. In the construction stage, the designer needs to oversee the construction of the design.

https://www.slideshare.net/SumanSaurabh9/srs-for-student-database-management-system

The link provided is a Software Requirements Specification (SRS) report for a Student Database Management System. The report outlines the functional requirements, system features, and user interfaces for the system.

The system is designed to maintain and process complete details of students about their personal, academic, sports, and health domains. It allows students to view their marks and grades anytime from their homes, enables professors to upload marks and submit grades from their offices, and allows medical officers and admin staff to view students' medical history and medical status in any case of emergency.

https://support.microsoft.com/en-us/office/database-design-basics-eb2159cf-1e30-401a-8084-bd4f9c9ca1f5

The link provided is a guide titled "Database Design Basics" from Microsoft Support. The guide provides an overview of the principles and steps involved in designing a database, focusing on a desktop database.

The guide begins by emphasizing the importance of good database design, which helps to reduce redundant data, provide access with the required information, and ensure the accuracy and integrity of data. It outlines two main principles of good database design: avoiding duplicate information (redundant data) and ensuring the correctness and completeness of information.

2.2 Interview

2.2.1

i. Group 3 : Interview Plan

System : Sanjay Studio

Participants: Garvika Singh Rajawat (Student - DAIICT)

Aditi Singh (Student - DAIICT)

Mr. Sanjay (Sanjay Studio)

Date: 14/09/2023

Time : 18:00

Duration : 40 minutes

Place: Sanjay's office

Purpose of Interview : The primary goal of our interview was to understand the needs, preferences, and challenges that interior designers face in managing their projects and to explore how a database management system could address these issues.

Agenda : Agenda was to understand the day to day operations of the interior designing industry and understand the entities involved and ask relevant questions in the given time to understand the functional and non-functional requirements of the firm.

Transcript of our conversation with Mr. Sanjay:

What are the key features that an interior design database management system should have?

As an interior designer, I believe the key features would include the ability to store and manage client information, project management functionalities, inventory and material tracking, integration with design software, and reporting and analytics capabilities. These would enable me to keep track of all project details in one central location, manage timelines and budgets effectively, and make informed decisions based on data.

How do you currently manage your interior design projects?

At present, I manage my projects using a mix of traditional and digital tools. For example, I use spreadsheets to track budgets and timelines, design software for creating design plans, and physical folders for storing client information and material samples.

How would a database management system improve your workflow and project management?

A database management system would centralize all the project information, making it easier for me to access and manage. It would eliminate the need for manual tracking of project details which currently takes up a lot of my time.

How do you currently store and manage client information?

Currently, I store client information in spreadsheets and email threads. While this method works, it's not the most secure or organized way to store sensitive information.

What challenges do you face with your current project management system?

Although a spreadsheet for managing data can be a little time consuming, it is good enough for me since the number of projects I take on is limited. But I believe as the workflow increases, It will require a centralized project tracking system to save time and increase efficiency.

What kind of data would you store in an interior design database management system?

The kind of data I would store includes client details, project timelines and budgets, design plans, and material inventory. I would also like to rate systems to rate the quality of vendors/suppliers as good, bad or poor based on their supplies for future reference.

What are the entities involved in the business?

The entities involved in the business are Suppliers, Laborers, Materials, Customers, design types, Vendors and Employees

What privilege would you like to give to your clients in the system?

Tracking the project progress, making financial transactions pertaining to the project, catalog of designs and feedback mechanism. A mechanism where feedback can be converted into a task would be a good idea.

What are the general client requirements?

Clients generally come with a very scattered sense of requirements. They usually have no vision. So, the first thing I ask them is their budget, to be able to provide them some vision.

How would you like to access the database? Via a desktop application, web browser, or mobile app?

I would prefer to access the database via a web browser. This would give me the flexibility to access the database from any device with an internet connection, whether I'm at the office, at a client's site, or working from home..

How important is data security to you in a database management system?

Data security is extremely important to me. As the database would contain sensitive client information, it's imperative that the information is protected from unauthorized access and data breaches.

Would you like the system to have integration capabilities with other software or tools you use? If so, which ones?

Yes, integration with other software would be very beneficial. In particular, I would like the system to integrate with design software like AutoCAD or Sketchup, as well as project management tools like Trello or Asana. This would create a seamless workflow, allowing me to easily transfer data between the different tools.

What kind of reports would you like the system to generate for you?

As an interior designer, I would find reports on project progress, budget status, and inventory levels incredibly useful. These reports would give me an overview of how

my projects are progressing and whether they are staying within budget. Also, I would like to track profit/loss from each project, if I am at a loss in a certain project I should be able to track the scope of recovery from the other projects.

Would you prefer a cloud-based or on-premise database system? Why?

I would prefer a cloud-based system. A cloud-based system is more accessible. As long as I have internet access, I can access the database from anywhere, which is not always possible with on-premise systems. Cloud-based systems usually offer more scalability. As my business grows, I can easily increase my storage capacity.

How important is ease of use to you in a database management system?

Ease of use is of utmost importance. A system that is difficult to use or understand can hinder work progress and lead to mistakes. In the past, I tried to implement a project management system to streamline workflow, but it proved difficult for my employees to access. The User Interface was too complicated to understand, which led to redundancy in work. Consequently, I had to revert to using Excel.

How would you like the system to handle data backup and recovery?

I would like the system to handle data backup and recovery automatically. Regular backups should be performed to prevent data loss, and in the event of any data loss, there should be a straightforward recovery process. This would give me peace of mind knowing that my data is safe and can be recovered if necessary.

How do you envision the database management system supporting your business growth?

As my business grows, I expect to take on more projects and clients, and the database system should be able to scale to accommodate this growth. It should also be flexible enough to support the introduction of new features or integrations as the needs of my business evolve. It should be easy to work with, and should require minimum training of personnels.

Would you like the system to have collaboration features, such as task assignment or sharing of information?

Yes, collaboration features would be very useful. In many projects, I work with a team of designers, contractors, and other professionals. Being able to assign tasks, share information, and communicate through the database system would facilitate teamwork and improve project coordination.

What are the potential challenges or concerns you foresee in implementing a new interior design database management system?

One potential challenge could be the learning curve associated with using a new system. It may take some time for me and my team to get familiar with the new system and use it efficiently. Another concern could be data migration from our current system to the new one, as there could be risks of data loss or corruption.

ii. Group 3 : Interview Summary

System : Sanjay Studio

Participants: Garvika Singh Rajawat (Student - DAIICT)

Aditi Singh (Student - DAIICT)

Mr. Sanjay (Sanjay Studio)

Date: 14/09/2023

Time : 18:00

Duration : 40 minutes

Place: Sanjay's office

Purpose of Interview : The primary goal of our interview was to understand the needs, preferences, and challenges that interior designers face in managing their projects and to explore how a database management system could address these issues.

Summary of Interview :

- Current Project Management: Mr. Sanjay currently uses a combination of traditional and digital tools, including spreadsheets, design software, and physical folders, to manage his projects.

- Client Information Management: Presently, client information is stored in spreadsheets and email threads, which Mr. Sanjay acknowledges is not the most secure or organized method.
- Challenges: Mr. Sanjay finds the current method of managing data via spreadsheets sufficient for his limited number of projects. However, he anticipates needing a more efficient system as his workload increases.
- Entities: The entities involved in the business include suppliers, laborers, materials, customers, design types, vendors, and employees.
- Client Privileges: He believes clients should be able to track project progress, make financial transactions, view design catalogs, and provide feedback within the system.
- Client Requirements: Mr. Sanjay mentioned that clients often come with a scattered sense of requirements, and budget is a crucial starting point for shaping their vision.
- Access: Mr. Sanjay prefers to access the database via a web browser, as it allows him to access the system from any device with an internet connection.
- Integration: Mr. Sanjay would like the system to integrate with design software like AutoCAD or Sketchup, and project management tools like Trello or Asana.
- Cloud-Based System: Mr. Sanjay prefers a cloud-based system due to its accessibility and scalability.

iii. List of requirements gathered from the Interview:

Requirement	Description
Client Information Management	The system should have the ability to securely store and manage client information. This would include contact details, project preferences, budget, and other

	relevant details.
Project Management:	The system could include features to track project timelines, budgets, and status. It should also provide a platform for communication between the designer and the client
Inventory and Material Tracking:	The system should allow for tracking of inventory and materials used in different projects.
Design Software Integration:	The system should be able to integrate with design software like AutoCAD or Sketchup. This would allow for seamless transfer of design plans and other related information
Reporting and Analytics:	The system should provide reporting and analytics capabilities. This would include reports on project progress, budget status, and inventory levels.
Vendor/Supplier Rating:	The system should allow Mr. Sanjay to rate vendors and suppliers based on their performance. This would help in future decision-making and supplier management.

Client Access	Clients should have the ability to track project progress, make financial transactions, view the catalog of designs, and provide feedback.
Cloud-Based System	Mr. Sanjay prefers a cloud-based system, as it offers more accessibility and scalability

2.3 Questionnaires

Client's Questionnaire

1- What is your time frame?(in months)

☐1

☐2

☐3

☐>3

2 - What is your budget for the project?

☐8L

☐Between(8L,12L)

☐Between(12L,25L)

☐>25L

3 - What is your preferred design style?

☐Contemporary

☐ Modern

☐ Traditional

☐ Minimalist

4 - Which colors do you prefer for your space?

☐ Creamy hues

☐ Bright colors

☐ Dark colors with warm undertone

☐ Pastels

☐ Other

5 - Which particular materials or textures would you like to use?

☐ Fabric texture

☐ Ceiling texture

☐ Textured walls

☐ Glass and mirror texture

6 - Do you have any preferences for window treatment?

☐ Curtain and draperies

☐ Blinds

☐ Shutters

☐ Shades and sheer curtain

☐ Motorized shades

7 - Are there any specific storage needs or organizational solutions you would like to incorporate?

☐ Build-in shelving and cabinets

☐ Closet systems

☐ Under stair storage system

☐ Wall mounted storage

☐ Open shelving

8 - What are your preferences for artwork?

☐ Abstract

☐ Landscape

☐ Nature

☐ Photographs

☐ Other

9 - What are your preferences for flooring?

☐ Hardware

☐ Tile

☐ Marble

☐ Concrete

☐ Carpet

10 - Do you have any specific requirements for lighting?

☐ Ambient lighting

☐ Task lighting

☐ Track lighting

☐ Accent lighting

☐ Sconce lighting

☐ Chandeliers

Designer's Questionnaire

1 - Which type of interior design project do you primarily work on?

☐ Commercial

☐ Residential

☐ Hospitality

☐ Other

2 - What software tools are you currently using for your interior design work?

☐ AutoCAD

☐ SketchUp

☐ Adobe Creative Suite

☐ Other

3 - What is your preferred platform for running the software?

☐ Windows

☐ Mac

☐ Both

☐ other

4 - Which features are essential for your interior design software?

☐ 2D/3D modeling

☐ Material and furniture libraries

☐ Collaboration tools

☐ Rendering capabilities

5 - How do you currently handle client communication and collaboration within your projects?

- ☐ Email
- ☐ Phone
- ☐ Project management software
- ☐ Dedicated client portal

6 - What file formats do you commonly use for design assets?

- ☐ JPEG/PNG
- ☐ DWG/DXF
- ☐ 3D models (e.g., OBJ, FBX)
- ☐ Other

7 - Do you need integration with other software or services in your workflow?

- ☐ CAD software
- ☐ Project management tools
- ☐ 3D modeling software
- ☐ None

8 - What pricing model do you prefer for software licensing?

- ☐ Subscription-based
- ☐ One-time purchase
- ☐ Free with limited features
- ☐ Other

9 - How important is user interface (UI) and user experience (UX) design to you in choosing software?

- ☐ Extremely important

☐ Important

☐ Somewhat important

☐ Not important

10 - How do you envision the software improving your interior design process?

☐ Saving time

☐ Streamlining collaboration

☐ Enhancing design quality

☐ Other

11 - Would you require technical support and training for your team to use the software effectively?

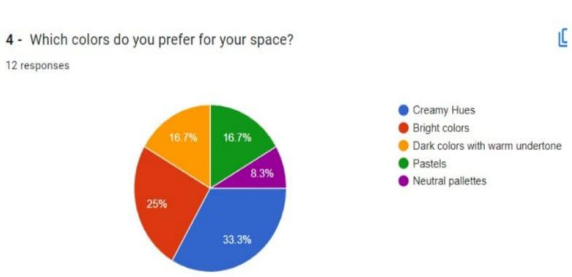
☐ Yes, extensive training and support

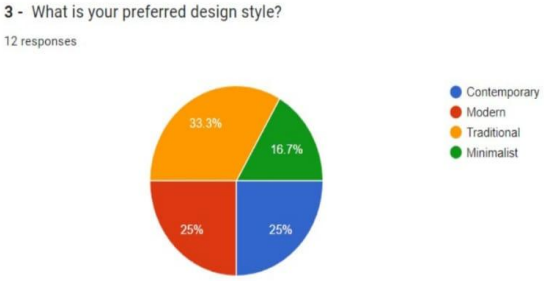
☐ Yes, basic training and support

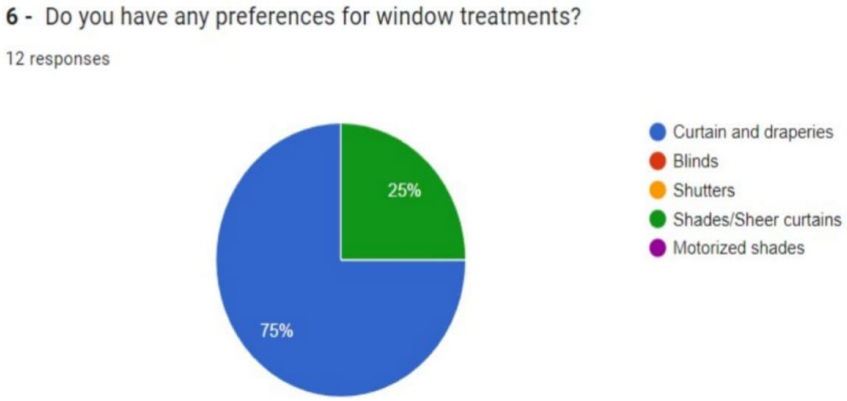
☐ No, self-sufficient

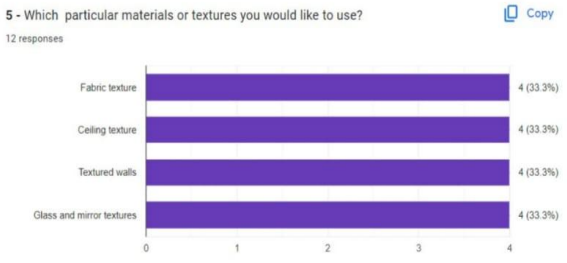
☐ Unsure

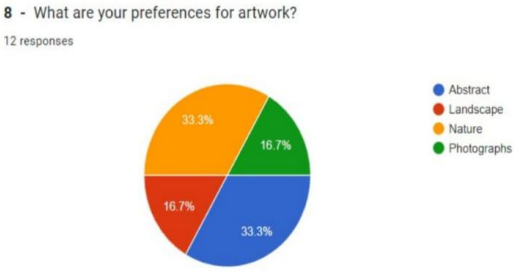
Client's Questionnaire summary :

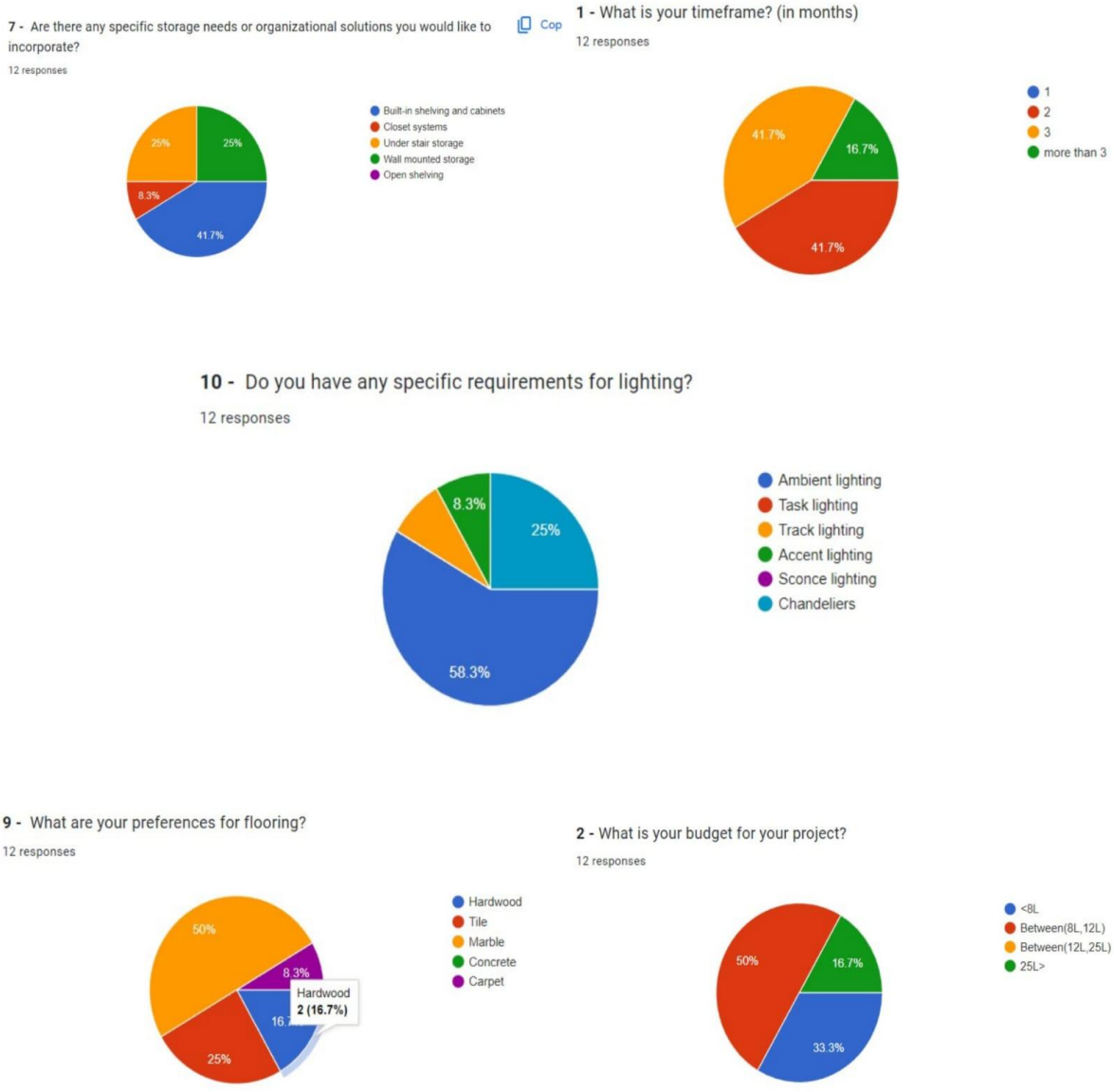










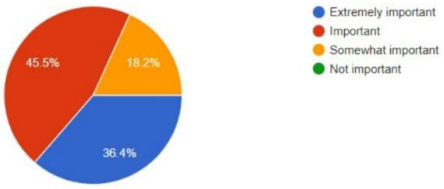


A sizable majority of people like soft and creamy color palettes in current home design trends, which reflects a taste for serene and welcoming environments. A third of people say that modern style appeals to them because it is sleek and simple. The majority of people favor draperies and curtains for window decorations. Additionally, a sizable 58% of consumers use ambient lighting options to improve the ambiance of their spaces. The majority of interior design clients want their projects to be finished quickly—within a two-month timeframe—so speed is essential. The need for practical yet fashionable design solutions is highlighted by the fact that many people are working within budgets of between 8 and 12 lakhs.

Interior designer's questionnaire summary :

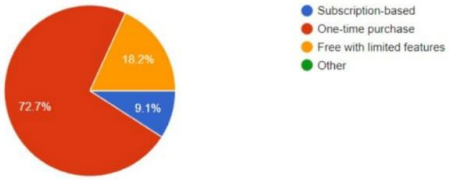
How important is user interface (UI) and user experience (UX) design to you in choosing software?

11 responses



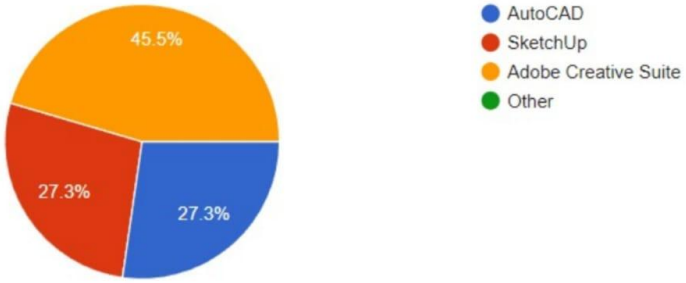
What pricing model do you prefer for software licensing?

11 responses



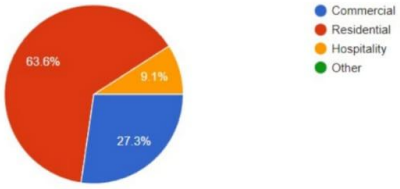
What software tools are you currently using for your interior design work?

11 responses



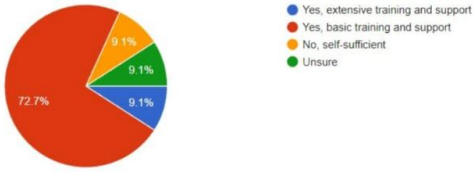
Which type of interior design project do you primarily work on?

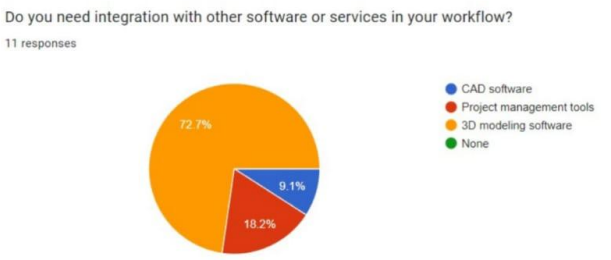
11 responses

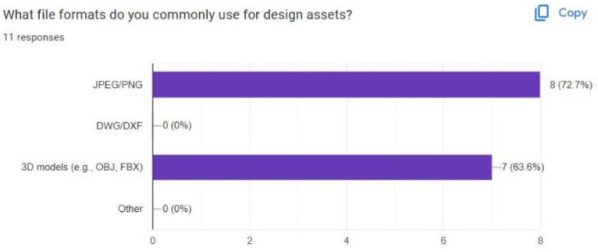


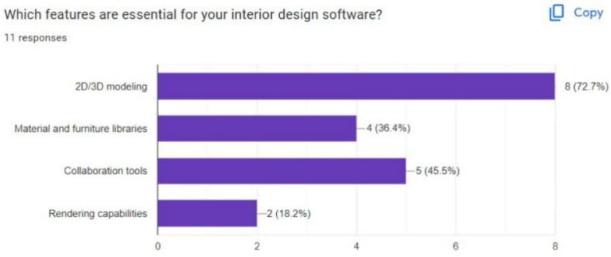
Would you require technical support and training for your team to use the software effectively?

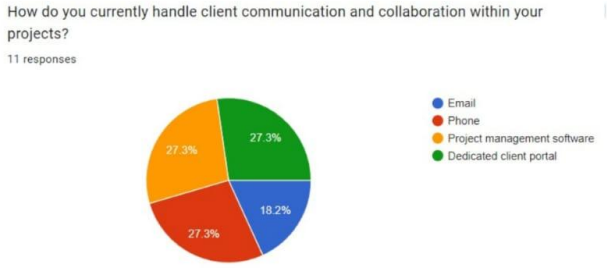
11 responses

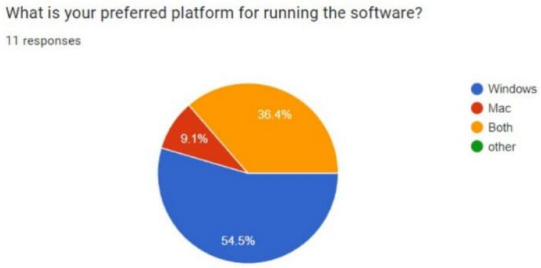


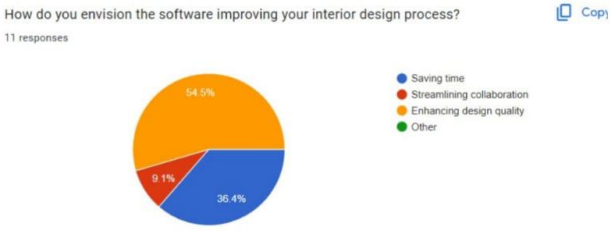












Most interior designers today prefer to use Adobe Creative Suite since it helps them produce far better work. Their primary focus is on residential developments, which account for around 64% of all of their projects. Tools that improve project management are highly desired by designers. For their design work, they enjoy utilizing images in formats like JPEG or PNG as well as 3D models. Additionally, the majority of designers use Windows-based computers. It's fairly typical in the field.

2.4 Observation

2.4.1. Summary

- Adobe Creative Suite is preferred by the majority of interior designers for better design work.
- With around 64% of their projects being residential constructions, this sector is their main focus.
- Project management solutions are highly valued by designers.
- In terms of picture formats, JPEG, PNG, and 3D models are preferred for design work.
- Interior designers frequently utilize Windows-based computers.
- Current home design trends embrace soft, milky color schemes that produce calming, welcoming spaces.
- A third of respondents say that the sleek, uncomplicated aesthetics of modern style appeal to them.
- The majority of individuals choose to decorate their windows with draperies and curtains.
- 58% of consumers use ambient lighting solutions to improve the mood of their spaces.
- Clients of interior designers often want their projects finished in two months.

2.4.2. Requirements -

1. Project information management
2. Budget tracking
3. Design inspiration and feedback
4. Material and product selection
5. Security and privacy
6. Accessibility and ease of use
7. Project timeline management

2.5 Fact Finding Chart

Objective	Technique	Subject(s)	Time Commitment
Understand the current data management system	Interview, observations	Interior designers,	1 hr
Identify problems with the current system	Questionnaire, observations	Interior designers,	1 hr
Gather requirements for the new system	Interview, Questionnaire, observations	Interior designers, Clients	2 hr
Understand the terminology used in the enterprise	Interview	Interior designers, Clients	1 hr
Identify opportunities from the new system	Interview, observations	Interior designers, Clients	1 hr
Understand necessary constraints on the data	Interview, observations	Interior designers, Office managers, IT team	1 hr
Gather a prioritized set of requirements for the new system	Interview, Questionnaire, observations	Interior designers, Office managers, Clients	2 hr
Understand the data flow	Observation, observations	Interior designers, Office managers, IT team	1 hr

3. List of Requirements

1. Project Management

Functionality	Description
Product Catalog Management	The system would enable administrators to add, update, or remove interior design products from the catalog. It should also allow categorization of products based on various factors like style, room type, color, etc
Ratings system	The system would allow interior designers to rate the quality of material supplied and vendor service as good, bad or poor.
Project track	The system would be able to track the status of the project and provide updates to customers. It should also allow administrators to manage and update project statuses
Reporting	This functionality will provide various reports to help users make informed decisions. It will include reports on project progress, financial status, inventory status, and client relationships.
Interior Design Project Management	The system would allow interior designers to manage their projects, including client

	communication, design drafts, revisions, and final approval.

2. Client Management

Project track	The system would be able to track the status of the project and provide updates to customers. It should also allow administrators to manage and update project statuses.
Project Reviews and Feedback	Users would be able to rate and review products. The system will be able to display these reviews and ratings on the project page for reference by other users.
Payment Processing	The system would securely process payments using various methods like credit cards, debit cards, or digital wallets. It should also handle refunds in case of cancellations or returns.

3. Inventory and supply

Supplier Management	This functionality will manage relationships with suppliers. It will include details about products supplied, delivery schedules, and payment terms.

4. Data Management

Data Security and Privacy	The system would ensure the security and privacy of user data and transactions.
Data Integrity	The system would maintain data integrity, especially in cases of denormalization, with mechanisms like batch processing, transaction control, or triggers.
User Access Management	This functionality will manage user access to the IDDBMS. It will define user roles and permissions to ensure that only authorized users can access certain functionalities or data. It will also track user activities for auditing and security purposes.

5. Reporting

Reporting and analytics capabilities	The system would have built-in reporting and analytics features to track project progress, performance, and financials. This would provide valuable insights for decision-making and project evaluation.

6. Integration

Collaboration features	The system would facilitate communication and collaboration between designers, clients, vendors, and stakeholders. This may include features such as document sharing, task assignment, and real-time progress tracking. Integration with communication tools like Slack or Microsoft Teams could enhance collaboration.
Integration with design software	The system would be able to import and export design files from popular design software such as AutoCAD or SketchUp. This integration would streamline the design process and ensure seamless data transfer.

4. User Categories & Privileges

4.1 List of Users

1. Administrator
2. Interior designer
3. Client
4. Supplier
5. Employee

4.2 Privileges of Users

User Role	Privileges
Administrator	Full access to all features of the system, including user management, project management, inventory management, and reporting. They can add, modify, and delete all data in the system
Interior Designer	Can create, update, and view projects they are assigned to. They can also view and update inventory and material information, and generate reports related to their projects.
Client	Can view projects they are involved in, make financial transactions, view design catalogs, and provide feedback. They can't modify project details or access other clients' data.
Supplier	Can view and update their own product details and prices, but can't access other suppliers' data or any project information.

User Role	Privileges
Administrator	Full access to all features of the system, including user management, project management, inventory management, and reporting. They can add, modify, and delete all data in the system
Employee	Can view project and client information, update inventory details, and perform tasks assigned by the interior designer. They can't modify project or client details

5. List of Assumptions and Dependencies

5.1 Assumptions

- **Data structure:**For storing and organizing information relevant to interior design, including furniture, materials, suppliers, clients, projects, etc., the database presupposes a specified structure.
- **Data integrity:** The database makes the supposition that the information input into the system is correct, comprehensive, and consistent. To ensure data integrity, it could also contain constraints and validation rules.
- **User roles and permission:**The database makes the assumption that various user roles—such as designers, administrators, and clients—exist and that they have the appropriate access rights to the data. To regulate user access, it might use authentication and authorisation procedures.
- **Reliability:**The database counts on its availability and dependability to be there for data storage and retrieval. To guarantee data integrity in the event of failures or disasters, it might contain backup and recovery techniques.

5.2 Dependencies

- **Hardware and software:**The database infrastructure, which includes servers, storage, networks, operating systems, and the database management system (DBMS) software itself, must be available and compatible for the database system to function.

- **Database design:**The database system relies on a carefully constructed database schema that accurately depicts the entities, relationships, and attributes important to interior design. To lessen duplication and increase data consistency, the data are normalized.

- **Application integration:** Integration of the database system with other workflow-related software or systems, such as CAD software, project management instruments, accounting systems, etc., may be necessary. Data sharing, synchronization, or interoperability may all be a part of this integration.

- **Data privacy and security:** In order to safeguard sensitive data, such as customer information, trade secrets, or intellectual property, the database system may rely on security procedures. This may involve data privacy laws compliance, access controls, audit trails, and encryption.

6. Business Constraints

1. **Data security :** Protecting sensitive client information and design concepts is crucial.Implementing robust security measures to prevent data breaches is essential.
2. **Privacy regulations :** Compliance with data protection regulations ,depending upon the nature of the data is mandatory.
3. **Performance :** Ensuring that the database can handle complex queries and large datasets.
4. **Data integrity :** Maintaining the accuracy and consistency of data,especially when multiple users are updating the data concurrently.
5. **Cost :** Budget constraints may impact the choice of technology and infrastructure for the database system.
6. **Backup and recovery :** Having a robust backup and recovery plan is crucial to avoid data loss.

II. ERD

1. Noun Analysis Table

1.1 Noun (& Verb) Analysis.

i. Find the nouns (entities) or verbs (relationships) in sentences of the problem description using Noun Analysis Method. 1. List all the extracted Nouns & Verbs in the below-given table format.

Nouns (Entities)	**Verbs (Relationships)**
Designer	Includes
Residential home	Transforms
Business	Outfitting
Client	Enlist
Color_Schemes	Choose
Flooring_materials	Give
Furniture	Establishes
Artwork	Outlining
Accessories	Defines
Area	Includes
Design	Will be able to
Software_Requirements_Specification (SRS)	Create
Functionalities	Monitor
Features	Visualize
Specifications	Provides
System	Allows
Development_team	Store
Stakeholders	Access
Project	Create and manage
Product_Scope	Assign
Design_ideas	Set
Vendor’s_information	Track
Client_management	Enables
Budget_expense_tracking	Share
Architect’s_data	Receive

Private_property	Iterate
Users	Provide
Furniture_supplies	Review
Quotations_invoices	Approve
Design_projects	Facilitates
Design_thoughts	Search
User_authentication_authorization	Compare
invoices	takes
Project_management	Place
Furniture_material_catalog	Update
Designer	makes
Design_visualization	Includes
Reporting_analytics	Estimate
User_collaboration	Track
Materials	uses
Interior_Designer	Generate
Client	Monitor
Vendor	Generate
Administrator	Manage
Vendor	assigned
Project_planning	Issue
Design_concept_development	Confirm
Material_sourcing	Putting
Budgeting	creates
Progress_tracking	managing
Project-related_information	
Client_requirements	
Design_specifications	
Supplier_details	

Project_timelines	
Design_Collaboration	
Supplier_Management	
Financial_Tracking	
Managing_calendar	

1.2. Create Table.1. & Table.2. as per the below-given format for accepted nouns list.

Nouns (Entities)	**Reasons**
Interior designer	Interior designer information is vital entity set
Client	Client information and details need to be stored for future use and is a vital entity set
Accessories	Can be considered as a entity set that includes various design types
Project	Includes details of the project the interior designer is working on.
Vendors	Entity set that contains vendor information for reference
Project Budget	Can be an attribute in the project entity set.
Budget and expense	Vital entity set that would contain financial information to keep track of expenses and budget in projects.
Architect	Includes details of the architects that interior designer contacts for various projects
Property	Can include the properties like warehouses and other store properties owned by designer
Users	Can include combined information of all the users of the system.
Reports	Can include reports of the budget, project track etc, can be a crucial entity sets
Administrator	The system admin info and the team that is part of the admin department, can either be an attribute to the employee entity or can be an entity set itself.

Supplier	Crucial entity set that contains information about the suppliers , the materials that are supplied by them

Noun	**Likely entity set to be assigned**
Client requirements	Client
Vendor’s information	Vendor
Design specifications	Designs
Design visualization	Designs
Design thoughts	Designs
Design projects	Designs
Design ideas	Designs
Color schemes	Designs
Project timelines	Project
Project deadline	Project
Project plan	Project
Project Budget	Project
Furniture	Accessories
Flooring material	Accessories
Art work	Accessories
Material sourcing	Supplier
Furniture and supplies	Supplier

1.3 Create Table.3. as per the below-given format for rejected nouns list.

Noun	**Rejection reason**
Vendor	its an repeated entity set
Design collaboration	it does not make sense for any entity set
Managing calendar	its neither an entity set nor an attribute.
User collaboration	its neither an entity set nor an attribute.
Development team	it is redundant attribute

Design concept development	its neither an entity set nor an attribute.
Design visualisation	its neither an entity set nor an attribute.
User authentication and authorisation	its neither an entity set nor an attribute.
Project planning	its neither an entity set nor an attribute.
Project management	it is redundant attribute
Project planning	it is redundant attribute
Progress tracking	it is redundant attribute
Financial tracking	its neither an entity set nor an attribute.
Supplier management	it is redundant attribute
Clients requirement	it can not be defined as an entity or attribute
budgeting	it can not be defined as an entity or attribute
Material sourcing	it can not be defined as an entity or attribute

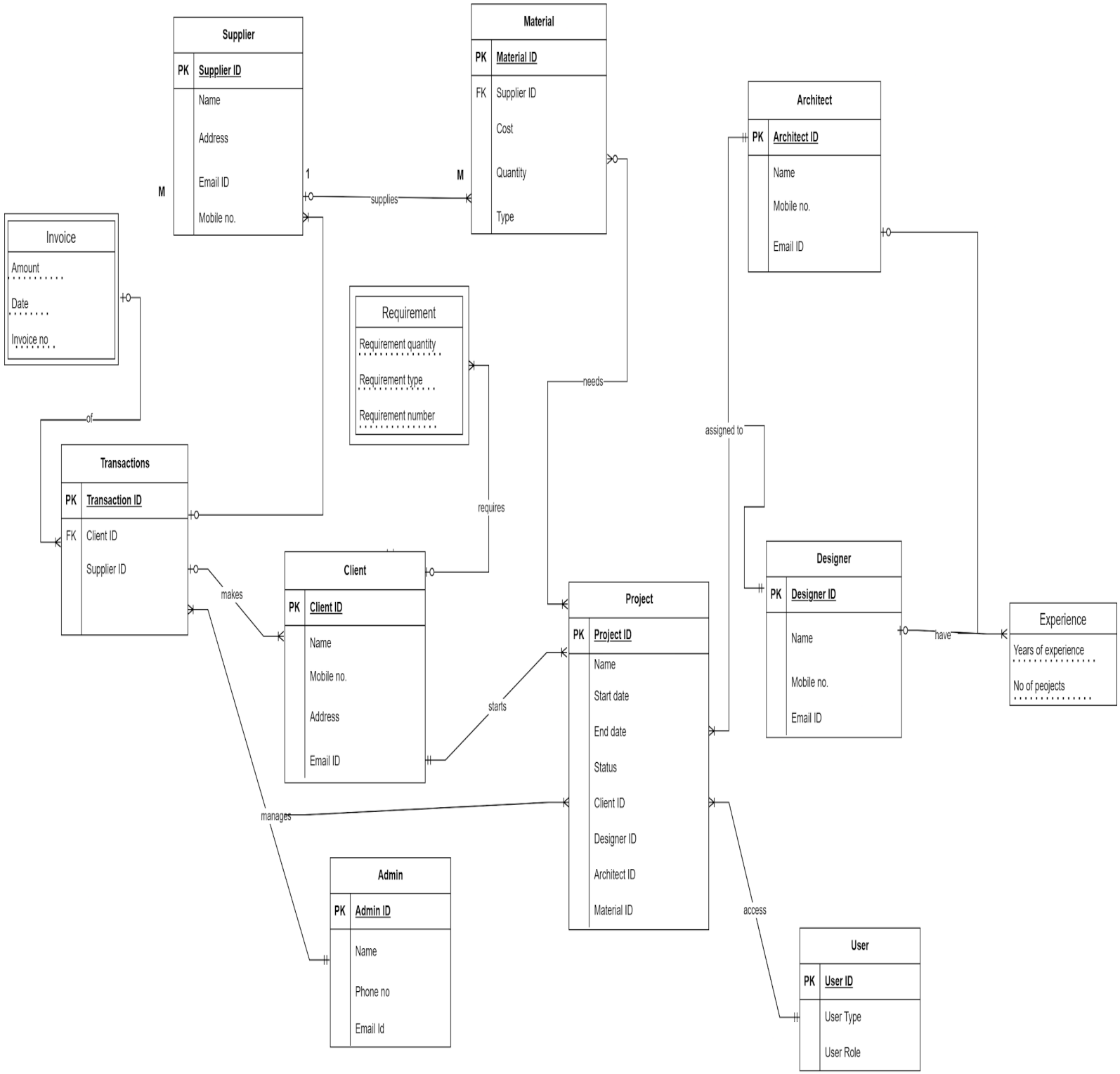
2. Entity-Attribute Table

Entity	**Attributes and Constraints**
Project	**PK** - Project ID **FK** - Client ID, Designer ID, Architect ID, Material ID **Not Null** - Name **Other-** Start Date, End Date, Status
Designer	**PK -** Designer ID **Not Null -** Name **Other -** Phone no, Email ID
Architect	**PK -** Architect ID **Not Null -** Name **Other -** Phone no, Email ID
Supplier	**PK -** Supplier ID **Not Null -** Name

	Other - Address, Phone no, Email ID
Material	**PK -** Material ID **FK -** Supplier ID **Not Null -** Name **Other -** Cost, quantity, type
Client	**PK -** Client ID **Not Null -** Name **Other -** Address, Phone no, Email ID
Transaction	**PK -** Transaction ID **FK -** Supplier ID, Client ID
Admin	**PK -** Admin ID **Not Null -** Name **Other -** Phone no, Email ID
User	**PK -** User ID **Not Null -** User role **Other -** User type

3. ERD

(next page)


Supplier
PK Supplier ID
Name
Address
Email ID
Mobile no.
M
1
supplies
M
Material
PK Material ID
FK Supplier ID
Cost
Quantity
Type
Architect
PK Architect ID
Name
Mobile no.
Email ID
Invoice
Amount
Date
Invoice no
of
Requirement
Requirement quantity
Requirement type
Requirement number
needs
assigned to
Transactions
PK Transaction ID
FK Client ID
Supplier ID
requires
makes
Client
PK Client ID
Name
Mobile no.
Address
Email ID
starts
Project
PK Project ID
Name
Start date
End date
Status
Client ID
Designer ID
Architect ID
Material ID
Designer
PK Designer ID
Name
Mobile no.
Email ID
have
Experience
Years of experience
No of peojects
manages
Admin
PK Admin ID
Name
Phone no
Email Id
access
User
PK User ID
User Type
User Role

III. Database Schema

1. Relational Schema

1. Project(Project_id, Client ID, Designer ID, Architect ID, Material ID, Admin ID, User ID, Name, Start Date, End Date, Status)
1. Designer(Designer_ID, Name, Phone no, Email ID)
2. Architect(Architect ID, Name, Phone no, Email ID)
3. Supplier(Supplier ID, Name, Address, Phone no, Email ID)
4. Material(Material ID, Supplier ID, Name, Cost, quantity, type)
5. Client (Client ID, Name, Address, Phone no, Email ID)
6. Transaction(Transaction ID, Supplier ID, Client ID)
7. Admin(Admin ID, Name, Phone no, Email ID)
8. User(User ID,User role,User type)

2. Schema Refinement Process

1. **Project(Project_id, Client ID, Designer ID, Architect ID, Material ID, Admin ID, User ID, Name, Start Date, End Date, Status,Rating)**
   There is no multi-valued attribute .
   Table is already in 1NF.
   Candidate key: {Project_id, Client ID, Designer ID, Architect ID, Material ID, Admin ID, User ID}

   Functional dependency: Project_id ~ Start Date, End Date, Status
   Client ID ~ Rating

   There is partial dependency of rating,start date and end date with the subset of candidate key
   Not in 2NF
   Decompose the given table as-

Table 1- Project(Project_id, Client ID, Designer ID, Architect ID, Material ID, Admin ID, User ID, Name)

Table 2- Project_details(Project_id,Start Date, End Date, Status)

Table 3- Client's_rating(Client_ID, Rating)

2. Designer(Designer_ID, Name, Phone_no, Email ID)

Candidate key: {Designer_ID}
This is in 1NF.
Candidate key: {Designer_ID}
Prime attribute: Designer_ID
Non-Prime attribute: name, Email_ID, Phone_no

All Non-Prime attributes are fully functional dependent on prime attribute .
This is already in 2NF.
There is no transitive dependency between attributes (no non-prime attribute
Is dependent on any other non-prime attribute)
This is already in 3NF

3. Supplier(Supplier ID, Name, Address, Phone no, Email ID, Rating)

Candidate key: {Supplier ID}
This is in 1NF
Functional dependency: Supplier ID ~ Rating
Candidate key: {Supplier ID}
Prime attributes: Supplier ID
Non-Prime attributes: Name, Address, Phone no, Email ID, Ratings
All Non-Prime attributes are fully functional dependent on prime attribute .
This is already in 2NF.
There is no transitive dependency between attributes (no non-prime attribute
Is dependent on any other non-prime attribute)
This is already in 3NF

4. Material(Material ID, Supplier ID, Name, Cost, quantity, type)

No multivalued attribute
Already in 1NF

Candidate key: Material ID, Supplier ID,Name, Type
Functional dependency: Supplier_ID ~ Cost, Material ID ~ quantity
Prime attributes: Material_ID, Supplier_ID, Name, Type

In this case there is functional dependency of cost AND quantity on subset of candidate key .This is not in 2NF

We need to follow decomposition -

Material(Material ID, Supplier ID, Name, type)
Material_cost(Supplier_ID,Cost)
Material_quantity(Material ID, quantity)

Now, this is in 2NF.

There is no transitive dependency

This is in 3NF

5. Client (Client ID, Name, Phone no, Email ID)

Candidate key: {Client ID}

This is in 1NF.

Prime attribute: Client ID
Non-Prime attribute: name, Email_ID, Phone_no, Client ID

All Non-Prime attributes are fully functional dependent on prime attribute .
This is already in 2NF.
There is no transitive dependency between attributes (no non-prime attribute
Is dependent on any other non-prime attribute)
This is already in 3NF

6. Transaction(Transaction ID, Supplier ID, Client ID)

There is no multivalued attribute.
This is already in 1NF.

7. **Admin(Admin ID, Name, Phone no, Email ID)**

Candidate key: {Admin ID}

Now this is in 1NF.
Candidate key: {Admin ID}
Prime attribute: Admin ID
Non-Prime attribute: name, Email_ID, Phone_no

All Non-Prime attributes are fully functional dependent on prime attribute .
This is already in 2NF.
There is no transitive dependency between attributes (no non-prime attribute
Is dependent on any other non-prime attribute)
This is already in 3NF

8. **User(User ID,User role,User type)**

Multi-valued attribute: User role

Not in 1NF

Decompose the table as:

User(User ID,,User type)
User_role(User ID,User role)

ANOMALIES

1. **Client----------Project (One to many)**

Insertion Anomaly: Any new client needs to be connected to a minimum of one project, you may run into issues when attempting to add them to the database. Data redundancy and even meaningless project entries may result from this.

Deletion Anomaly: If you accidentally remove a client from the database, you may also

accidentally remove all of the projects that are related to that customer. Data related to other clients' projects may be lost as a result of this.

Update Anomaly: If a project is linked to several customers, you may need to update information about the project in multiple records. If not managed appropriately, this can result in discrepancies.

2. **Architect----------Project (One to many)**

Insertion Anomaly: If a project has to be linked to an architect, you may run into issues when attempting to enter it into the database. If not handled carefully, you can end yourself having to make an architect placeholder in order to add the project details. Redundant data and maybe meaningless architect entries may result from this.

Deletion Anomaly: You may inadvertently erase all related projects if you remove an architect from the database. Project data related to other architects may be lost as a result of this.

Update Anomaly: If a project has numerous architects affiliated with it, you may need to update information about the project in multiple entries. If not handled properly, this could result in inconsistent results.

3. **Designer----------Project (One to many)**

Insertion anomaly: If you don't have a matching designer ID to link to the project, you may encounter issues when trying to add a new project to the database because every project needs to be connected to a designer. This may compel you to enter an empty designer field or create a placeholder designer, which may result in inconsistent data and possibly meaningless designer entries.

Deletion anomaly: You might erase every project connected to a designer if you choose to remove them from the database. Project data that could be connected to other designers is lost as a result of this. For instance, you might lose all the project records if Designer A worked on several projects and you were to delete Designer A.

Updation anomaly: If a designer is connected to several projects, you might need to update several project records when you need to update information about the designer (such as the designer's contact details). If you don't accurately update all related project records, this could result in inconsistent data.

4. **User----------Project (One to many)**

Insertion Anomaly:If you want to insert a new project into the database, and each project is required to be associated with a user, you might encounter difficulties if you don't have a corresponding user ID to link to the project. This can force you to create a placeholder user or leave the user field empty, which can lead to data inconsistency and potentially meaningless user entries.

Deletion Anomaly: If you decide to delete a user from the database, you could unintentionally delete all the projects associated with that user. This results in the loss of project data that might be associated with other users. For example, if User A is associated with multiple projects, deleting User A might lead to the loss of those project records.

Update Anomaly: When you need to update information about a user (changing the user's role or type), you may need to update multiple project records if the same user is associated with different projects. This can lead to inconsistencies if you don't update all related project records correctly.

5. **Supplier----------Material(One to many)**

Insertion Anomaly: If you want to insert a new material into the database, and each material is required to be associated with a supplier, you might encounter difficulties if you don't have a corresponding supplier ID to link to the material. This can force you to create a placeholder supplier or leave the supplier field empty, which can lead to data inconsistency and potentially meaningless supplier entries.

Deletion Anomaly:If you decide to delete a supplier from the database, you could unintentionally delete all the materials associated with that supplier. This results in the loss of material data that might be associated with other suppliers. For example, if Supplier A is associated with multiple materials, deleting Supplier A might lead to the loss of those material records.

Update Anomaly: When you need to update information about a supplier (e.g., changing the supplier's address or contact details), you may need to update multiple material records if the same supplier is associated with different materials. This can lead to inconsistencies if you don't update all related material records correctly.

6. **Client----------Transaction(One to many)**

Insertion Anomaly: If you want to insert a new transaction into the database, and each transaction is required to be associated with a client, you might encounter difficulties if you don't have a corresponding client ID to link to the transaction. This can force you to create a placeholder client or leave the client field empty, which can lead to data inconsistency and potentially meaningless client entries.

Deletion Anomaly: If you decide to delete a client from the database, you could unintentionally delete all the transactions associated with that client. This results in the loss of transaction data that might be associated with other clients. For example, if Client A is associated with multiple transactions, deleting Client A might lead to the loss of those transaction records.

Update Anomaly: When you need to update information about a client (changing the client's address or contact details), you may need to update multiple transaction records if the same client is associated with different transactions. This can lead to inconsistencies if you don't update all related transaction records correctly.

3. Normalized Schema

RELATION - PROJECT

Field Name	**Data Type**	**Description**	**Constraints**
Project_id,	VARCHAR(10)	Identifier for Project	PRIMARY KEY
Client ID	VARCHAR(10)	Identifier for Client	CANDIDATE KEY
Designer ID	VARCHAR(10)	Identifier for designer	CANDIDATE KEY
Architect ID	VARCHAR(10)	Identifier for Architect	CANDIDATE KEY
Material ID	VARCHAR(10)	Identifier for Material	CANDIDATE KEY
Admin ID	VARCHAR(10)	Identifier for Admin	CANDIDATE KEY
User ID	VARCHAR(10)	Identifier for User	CANDIDATE KEY
Name	TEXT	Name of project	NOT NULL

RELATION - PROJECT-DETAILS

Field Name	Data Type	Description	Constraints
Project_id	VARCHAR(10)	Identifier for Project	FOREIGN KEY
Start Date	DATE	Starting date of a project	
End Date	DATE	End date of a project	
Status	INTEGER	Status	

RELATION- CLIENT'S RATING

Field Name	Data Type	Description	Constraints
Client_ID	VARCHAR(10)	Identifier for Client	FOREIGN KEY
Rating	FLOAT	Rating by a client	

RELATION - DESIGNER

Field Name	Data Type	Description	Constraints
Designer_ID	VARCHAR(10)	Identifier for designer	Primary key
Name	TEXT	Name of designer	NOT NULL

Email_ID	VARCHAR(30)	Email_id of designer	NOT NULL
Phone_no	INTEGER	Phone_no. Of designer	NOT NULL
Years of Experience	INTEGER	No of years of past experience	
No of Projects	INTEGER	No of past projects completed	

RELATION - SUPPLIER

Field Name	**Data Type**	**Description**	**Constraints**
Supplier ID	VARCHAR(10)	Identifier for supplier	Primary key
Name	TEXT	Name of supplier	NOT NULL
Email ID	VARCHAR(30)	Email_id of supplier	
Phone_no	INTEGER	Phone_no. Of supplier	

RELATION - MATERIAL

Field Name	**Data Type**	**Description**	**Constraints**
Materia_ID,	VARCHAR(10)	Identifier for material	PRIMARY KEY

Supplier ID	VARCHAR(10)	Identifier for material	PRIMARY KEY
Name	TEXT	Name of material	NOT NULL
type	TEXT	Type of material	

RELATION - MATERIAL_COST

Field Name	Data Type	Description	Constraints
Material_ID,	VARCHAR(10)	Identifier for material	FOREIGN KEY
Cost	INTEGER	Cost of material	

RELATION - MATERIAL-QUANTITY

Field Name	Data Type	Description	Constraints
Supplier ID	VARCHAR(10)	Identifier for material	FOREIGN KEY
Quantity	INTEGER	Quantity of material	NOT NULL

RELATION - CLIENT

Field Name	Data Type	Description	Constraints
Client ID	VARCHAR(10)	Identifier for Client	PRIMARY KEY

Name	TEXT	Name of client	NOT NULL
Email ID	VARCHAR(30)	Email-id of client	
Address	VARCHAR(50)	Address of client	NOT NULL
Phone_no	INTEGER	Phone_no. Of client	NOT NULL

RELATION - ADMIN

Field Name	**Data Type**	**Description**	**Constraints**
Admin ID	VARCHAR(10)	Identifier for Admin	PRIMARY KEY
Name	TEXT	Name of the admin	NOT NULL
Email ID	VARCHAR(30)	Email-id of admin	
Phone no.	INTEGER	Phone no. of admin	

RELATION - TRANSACTION

Field Name	**Data Type**	**Description**	**Constraints**
Transaction ID	VARCHAR(10)	Identifier for Transaction	PRIMARY KEY

Supplier ID	VARCHAR(10)	Identifier for Supplier	
Client ID	VARCHAR(10)	Identifier for Client	
Amount	BIGINT	Amount paid	Not Null
Invoice No	BIGINT	Invoice number	
Date	DATE	Date of transaction	

RELATION - USER

Field Name	**Data Type**	**Description**	**Constraints**
User ID,	VARCHAR(10)	Identifier for User	PRIMARY KEY
User type	TEXT	Describes the type of user	NOT NULL

RELATION - USER ROLE

Field Name	**Data Type**	**Description**	**Constraints**
User ID,	VARCHAR(10)	Identifier for User	FOREIGN KEY
User role	TEXT	Describes the role of user	NOT NULL

IV. DDL and SQL

1. DDL Script

```
CREATE TABLE Project(
   Project_id VARCHAR(10) PRIMARY KEY,
   Budget INTEGER,
   Client_ID VARCHAR(10),
   Designer_ID VARCHAR(10),
   Architect_ID VARCHAR(10),
   Material_ID VARCHAR(10),
   Admin_ID VARCHAR(10),
   User_ID VARCHAR(10),
   Name TEXT NOT NULL
);
```

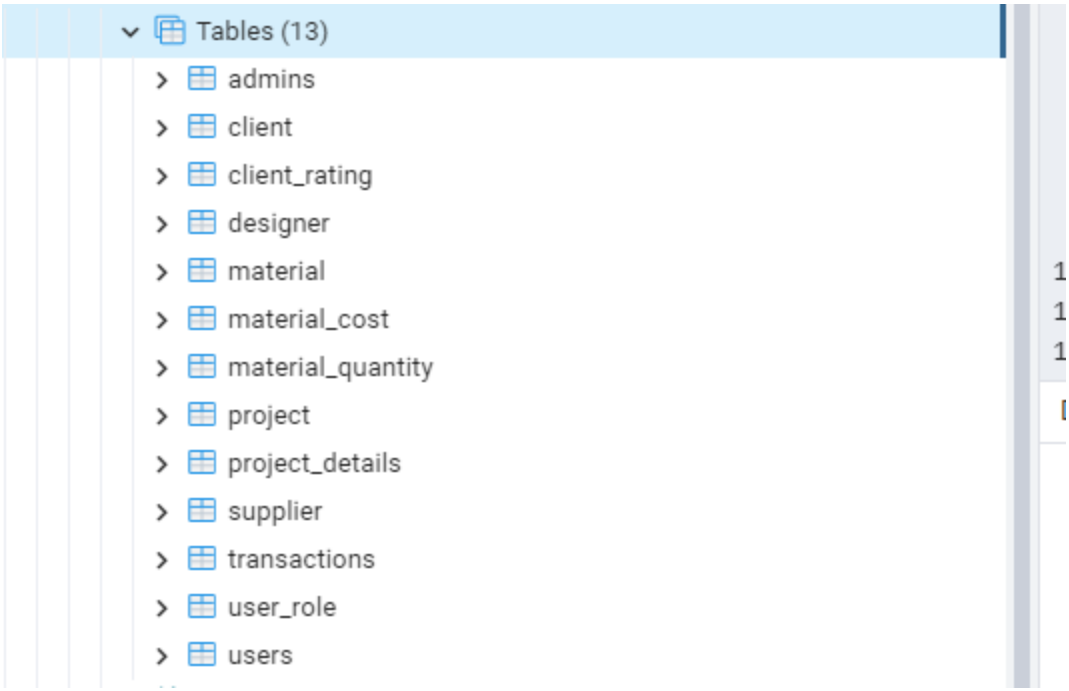
```
CREATE TABLE Architect(
   Architect_id VARCHAR(10) PRIMARY KEY,
   Name TEXT NOT NULL,
   Email_ID VARCHAR(30) NOT NULL,
   Phone_no INTEGER NOT NULL,
   Years_of_Experience INTEGER,
   No_of_Projects INTEGER
);
```

```
CREATE TABLE PROJECT_DETAILS(
  Project_id VARCHAR(10),
  Start_Date DATE,
  End_Date DATE,
```

```
  Status INTEGER,
  FOREIGN KEY (Project_id) REFERENCES PROJECT(Project_id)
);



CREATE TABLE CLIENT(
  Client_ID VARCHAR(10) PRIMARY KEY,
  Name TEXT NOT NULL,
  Email_ID VARCHAR(30),
  Address VARCHAR(50) NOT NULL,
  Phone_no INTEGER NOT NULL
);



CREATE TABLE CLIENT_RATING(
  Client_ID VARCHAR(10),
  Rating FLOAT,
  FOREIGN KEY (Client_ID) REFERENCES CLIENT(Client_ID)
);



CREATE TABLE DESIGNER(
  Designer_ID VARCHAR(10) PRIMARY KEY,
  Name TEXT NOT NULL,
  Email_ID VARCHAR(30) NOT NULL,
  Phone_no INTEGER NOT NULL,
  Years_of_Experience INTEGER,
  No_of_Projects INTEGER
);
```

```
CREATE TABLE SUPPLIER(
  Supplier_ID VARCHAR(10) PRIMARY KEY,
  Name TEXT NOT NULL,
  Email_ID VARCHAR(30),
  Phone_no INTEGER
);

CREATE TABLE MATERIAL(
  Material_ID VARCHAR(10),
  Supplier_ID VARCHAR(10),
  Name TEXT NOT NULL,
  type TEXT,
  PRIMARY KEY (Material_ID, Supplier_ID)
);

CREATE TABLE MATERIAL_COST(
  Material_ID VARCHAR(10),
  Supplier_ID VARCHAR(10),
  other_column TEXT,
  FOREIGN KEY (Material_ID, Supplier_ID) REFERENCES
MATERIAL(Material_ID, Supplier_ID)
);

CREATE TABLE MATERIAL_QUANTITY(
  Material_ID VARCHAR(10),
  Supplier_ID VARCHAR(10),
  Quantity INTEGER NOT NULL,
  FOREIGN KEY (Material_ID, Supplier_ID) REFERENCES
MATERIAL(Material_ID, Supplier_ID)
);
```

```
CREATE TABLE ADMINS(
  Admin_ID VARCHAR(10) PRIMARY KEY,
  Name TEXT NOT NULL,
  Email_ID VARCHAR(30),
  Phone_no INTEGER
);

CREATE TABLE TRANSACTIONS(
  Transaction_ID VARCHAR(10) PRIMARY KEY,
  Supplier_ID VARCHAR(10),
  Client_ID VARCHAR(10),
  Amount BIGINT NOT NULL,
  Invoice_No BIGINT,
  Date DATE
);

CREATE TABLE USERS(
  User_ID VARCHAR(10) PRIMARY KEY,
  User_type TEXT NOT NULL
);

CREATE TABLE USER_ROLE(
  User_ID VARCHAR(10),
  User_role TEXT NOT NULL,
  FOREIGN KEY (User_ID) REFERENCES USERS(User_ID)
);
```



2. Plain Text and SQL Queries

1. Retrieve the list of all suppliers along with their phone number.	SELECT Supplier_ID, Name, Phone_no FROM SUPPLIER;

2. Retrieve the list of all the designers along with their years of experience and number of projects they have worked on.	SELECT Designer_ID, Name, Years_of_Experience, No_of_Projects FROM DESIGNER;

3. Retrieve the list of all materials along with their suppliers.	SELECT m.Material_ID, m.Name AS Material_Name, m.type AS Material_Type, s.Supplier_ID,

	s.Name AS Supplier_Name FROM MATERIAL m JOIN SUPPLIER s ON m.Supplier_ID = s.Supplier_ID;

4. Retrieve the list of all the projects along with the designers name working on them	SELECT p.Project_id, d.Name AS Designer_Name FROM PROJECT p JOIN DESIGNER d ON p.Designer_ID = d.Designer_ID;

5. Count the number of designers per project.	SELECT p.Project_id, COUNT(d.Designer_ID) AS Number_of_Designers FROM PROJECT p LEFT JOIN DESIGNER d ON p.Designer_ID = d.Designer_ID GROUP BY p.Project_id;

6. Create a view that displays suppliers name and their phone number.	CREATE VIEW SupplierContactInfo AS SELECT Supplier_ID, Name AS Supplier_Name, Phone_no FROM SUPPLIER;

7. Create a view that displays the project id and their rating.	CREATE VIEW ProjectRating AS SELECT p.Project_id, cr.Rating FROM PROJECT p

	LEFT JOIN CLIENT_RATING cr ON p.Client_ID = cr.Client_ID; select * from projectrating;

8. List all the projects by 2023-01-01.	SELECT project_id, end_date FROM project_details WHERE end_date < '2023-01-01';

9. Retrieve the designers name who worked on a maximum number of projects.	SELECT d.Name AS Designer_Name FROM DESIGNER d WHERE (SELECT COUNT(*) FROM PROJECT p WHERE p.Designer_ID = d.Designer_ID) = (SELECT MAX(project_count) FROM (SELECT Designer_ID, COUNT(*) AS project_count FROM PROJECT GROUP BY Designer_ID) AS project_counts);

10. List projects in ascending order of their start date.	SELECT Project_id, Start_Date FROM PROJECT_DETAILS ORDER BY Start_Date ASC;

11. Create a view that displays the material type and their suppliers.	CREATE VIEW MaterialSupplier AS SELECT m.type AS Material_Type, m.Material_ID, m.Supplier_ID, s.Name AS Supplier_Name FROM MATERIAL m JOIN SUPPLIER s ON m.Supplier_ID = s.Supplier_ID;

	select * from materialsupplier;

12. List all the transactions greater than $650.	SELECT transaction_id, amount FROM transactions WHERE amount > '$650';

13. Fetch all the clients with their respective ratings:	SELECT CLIENT.Client_ID, CLIENT.Name, CLIENT_RATING.Rating FROM CLIENT INNER JOIN CLIENT_RATING ON CLIENT.Client_ID = CLIENT_RATING.Client_ID;

14. Fetch the details of the project with the longest duration:	SELECT PROJECT.Project_id, (PROJECT_DETAILS.End_Date - PROJECT_DETAILS.Start_Date) as Duration_in_days FROM PROJECT INNER JOIN PROJECT_DETAILS ON PROJECT.Project_id = PROJECT_DETAILS.Project_id ORDER BY Duration_in_days DESC LIMIT 1;

15. Fetch the details of the supplier providing the most number of materials:	SELECT material.Supplier_ID, material.type, COUNT(Material_ID) as Number_of_Materials FROM MATERIAL GROUP BY MATERIAL.Supplier_ID, material.type ORDER BY Number_of_Materials DESC LIMIT 1;

16. Fetch the details of the most expensive material:	SELECT MATERIAL.Material_ID, MATERIAL.type, MATERIAL_COST.Price FROM MATERIAL INNER JOIN MATERIAL_COST ON MATERIAL.Material_ID = MATERIAL_COST.Material_ID ORDER BY MATERIAL_COST.Price DESC LIMIT 1;

17. Fetch the details of the admin who has supervised the least projects:	SELECT Admin_ID, COUNT(Project_id) as Number_of_Projects FROM PROJECT GROUP BY Admin_ID ORDER BY Number_of_Projects ASC LIMIT 1;

18. Fetch the details of the admin who has supervised the most projects and also fetch the admin name:	SELECT A.Admin_ID, A.Name AS Admin_Name, P.Number_of_Projects FROM ADMINS A JOIN (SELECT Admin_ID, COUNT(Project_id) as Number_of_Projects FROM PROJECT GROUP BY Admin_ID) P ON A.Admin_ID = P.Admin_ID ORDER BY Number_of_Projects DESC LIMIT 1;

19. Find Architects with Highest Experience:	SELECT A.Architect_id, A.Name AS Architect_Name, A.Years_of_Experience FROM Architect A WHERE A.Years_of_Experience = (SELECT MAX(Years_of_Experience) FROM Architect);

20. List Suppliers and Their Materials along with Material Quantities and also create a view for the same and load results:	CREATE VIEW SupplierMaterialQuantity AS SELECT S.Supplier_ID, S.Name AS Supplier_Name, M.Material_ID, M.Name AS Material_Name, MQ.Quantity FROM SUPPLIER S JOIN MATERIAL M ON S.Supplier_ID = M.Supplier_ID JOIN MATERIAL_QUANTITY MQ ON M.Material_ID = MQ.Material_ID AND M.Supplier_ID = MQ.Supplier_ID; select * from suppliermaterialquantity;

21. List Projects with Designers Having More Projects Than Average:	SELECT P.Project_id, P.Client_ID, P.Designer_ID
	FROM PROJECT P
	WHERE P.Designer_ID IN (
	  SELECT Designer_ID
	  FROM DESIGNER
	  WHERE No_of_Projects > (
	  SELECT AVG(No_of_Projects)
	  FROM DESIGNER
	  )
	);

22. Calculate Average Material Quantity for Each Material:	SELECT M.Material_ID, AVG(MQ.Quantity) AS Avg_Quantity FROM MATERIAL M JOIN MATERIAL_QUANTITY MQ ON M.Material_ID = MQ.Material_ID AND M.Supplier_ID = MQ.Supplier_ID GROUP BY M.Material_ID;

23. Fetch details of project cost with	SELECT P.Project_id,

	M.type AS Material_type, MC.Price AS Material_Cost FROM PROJECT P JOIN MATERIAL M ON P.Material_ID = M.Material_ID AND P.Supplier_ID = M.Supplier_ID JOIN MATERIAL_COST MC ON P.Material_ID = MC.Material_ID AND P.Supplier_ID = MC.Supplier_ID;

24. List all ongoing projects.	SELECT Project_id, Start_Date, End_Date FROM PROJECT_DETAILS WHERE Start_Date <= CURRENT_DATE AND End_Date >= CURRENT_DATE;

25. Which project took the least amount of time.	SELECT Project_id, (End_Date - Start_Date) AS Duration FROM PROJECT_DETAILS ORDER BY Duration LIMIT 1;

26. List of all the materials.	SELECT Material_ID, Name, Type FROM MATERIAL;

27. List of all the clients and their names.	SELECT client_id, name FROM CLIENT;

28. Calculate the total number of suppliers.	SELECT COUNT(*) AS total_suppliers FROM SUPPLIER;

29. Calculate the number of materials used in each project based on type.	SELECT p.Project_id, m.type, SUM(mq.Quantity) AS total_quantity FROM

	PROJECT p JOIN MATERIAL m ON p.Material_ID = m.Material_ID JOIN MATERIAL_QUANTITY mq ON m.Material_ID = mq.Material_ID GROUP BY p.Project_id, m.type ORDER BY p.Project_id, m.type;

30. Retrieve the list of all the transactions along with the invoice number.	SELECT Transaction_ID, Invoice_No FROM TRANSACTIONS;

31. Create a stored procedure for adding a new project to the interior designing database.	CREATE OR REPLACE PROCEDURE AddNewProject(IN p_Project_id VARCHAR(100), IN p_Client_ID VARCHAR(100), IN p_Designer_ID VARCHAR(100), IN p_Architect_ID VARCHAR(100), IN p_Material_ID VARCHAR(100), IN p_Admin_ID VARCHAR(100), IN p_User_ID VARCHAR(100), IN p_Start_Date DATE, IN p_End_Date DATE, IN p_Status VARCHAR(100)) AS $$ BEGIN -- Insert data into PROJECT table INSERT INTO PROJECT(Project_id, Client_ID, Designer_ID, Architect_ID, Material_ID, Admin_ID, User_ID) VALUES (p_Project_id, p_Client_ID, p_Designer_ID, p_Architect_ID, p_Material_ID, p_Admin_ID, p_User_ID);

	-- Insert data into PROJECT_DETAILS table INSERT INTO PROJECT_DETAILS(Project_id, Start_Date, End_Date, Status) VALUES (p_Project_id, p_Start_Date, p_End_Date, p_Status); END; $$ LANGUAGE plpgsql;

32. Find the material supplied by each supplier along with its price and quantity?	SELECT S.Name AS Supplier_Name, M.Name AS Material_Name, MC.Price, MQ.Quantity FROM SUPPLIER S JOIN MATERIAL_COST MC ON S.Supplier_ID = MC.Supplier_ID JOIN MATERIAL M ON MC.Material_ID = M.Material_ID JOIN MATERIAL_QUANTITY MQ ON M.Material_ID = MQ.Material_ID AND S.Supplier_ID = MQ.Supplier_ID;

33. List the designers who have not worked on any project.	SELECT D.Name AS Designer_Name FROM DESIGNER D LEFT JOIN PROJECT P ON D.Designer_ID = P.Designer_ID WHERE P.Project_id IS NULL;

34. Fetch all the projects that ended before '2023-01-01'.	SELECT project_id, start_date, end_date FROM project_details WHERE end_date < '2023-01-01';

35. Fetch projects by ascending order of start_date	SELECT project_id, start_date FROM project_details ORDER BY start_date ASC;

36. Fetch project_id along with rating.	SELECT t1.project_id, t2.rating FROM project AS t1 INNER JOIN client_rating AS t2 ON t1.client_id = t2.client_id;

37. Fetch details of project by the amount paid by client for the respective project.	SELECT t1.project_id, t2.amount FROM project AS t1 INNER JOIN transactions AS t2 ON t1.client_id = t2.client_id;

38. Fetch details of a designer who has worked on the maximum no. of projects in the past.	SELECT *
	FROM designer
	WHERE no_of_projects = (
	  SELECT MAX(no_of_projects)
	  FROM designer
	);

39. Fech project details of a particular designer.	SELECT project_id FROM project WHERE designer_id = '01HEDA9WSW8PFQF6BKKE1YHF1F';

40. Fetch details of all the designers and their past year of experience and no. of projects.	SELECT designer_id, no_of_projects, years_of_experience FROM designer;

41. Trigger function to ensure every newly added designer has some experience.	CREATE OR REPLACE FUNCTION
	check_designer_experience()
	RETURNS TRIGGER AS $$
	BEGIN
	  IF NEW.Years_of_Experience <= 0 THEN
	  RAISE EXCEPTION 'Designer must have
	experience greater than 0 years';
	  END IF;
	  RETURN NEW;
	END;

	$$ LANGUAGE plpgsql; CREATE TRIGGER check_designer_experience_trigger BEFORE INSERT ON DESIGNER FOR EACH ROW EXECUTE FUNCTION check_designer_experience();

42. Fetch all projects along with their start date and end date.	SELECT P.Project_id, PD.Start_Date, PD.End_Date FROM PROJECT P JOIN PROJECT_DETAILS PD ON P.Project_id = PD.Project_id;

3. Executed Queries Outputs

1. Retrieve the list of all suppliers along with their phone number.	SELECT Supplier_ID, Name, Phone_no FROM SUPPLIER;

	supplier_id [PK] character varying (100)	name text	phone_no bigint
1	01HEDAH4BS7M8PTAG97TJKYXW7	Shaylynn Saiz	2208371050
2	01HEDAH4BTT1NG84RKMV8Y8670	Rodolphe Oldrey	4080974584
3	01HEDAH4BVH0KX5TD2JGB9YT2N	Heida Frewer	5373371886
4	01HEDAH4BWTV39ZX8T33FH1J01	Hyacinth Scowcroft	6471855990
5	01HEDAH4BXMFTS70C4G85BJJM1	Iona Abbey	7786151808
6	01HEDAH4BYTEP3K5QPQP3Z3T2R	Aviva Suthren	5030938679
7	01HEDAH4BYQR5W1DTH2DB3M8...	Hildegaard Castleman	3522931209
8	01HEDAH4BYT96P5K0PQ5V3NWYF	Evaleen Castellet	6637394605
9	01HEDAH4BZM3FMKKSQQTJYV8...	Leonhard Pottle	3046899685
10	01HEDAH4BZMWWFPPXTTFRMEA...	Simon Willisch	3330087141
11	01HEDAH4C0N4196R3RZ63VBZ22	Flin Burrows	5839304774
12	01HEDAH4C0XW2PYENHRW9YKW...	Robenia Bassham	5402942616
13	01HEDAH4C1RT6HZPJAXT5WTCPY	Cathrine Scarratt	4896802931

2. Retrieve the list of all the designers along with their years of experience and number of projects they have worked on.	SELECT Designer_ID, Name, Years_of_Experience, No_of_Projects FROM DESIGNER;

	designer_id [PK] character varying (100)	name text	years_of_experience bigint	no_of_projects bigint
1	01HEDA9WSDH8A9C51AD91R29DF	Violet Grafhom	25	67
2	01HEDA9WSFP80QC604G0FZ5MGR	Gussie Sebert	16	31
3	01HEDA9WSGY0PHY59P06XREZQ0	Gustav Harkness	1	36
4	01HEDA9WSHH0Z37QMA8BNRP464	Jackelyn Petegrew	16	2
5	01HEDA9WSKQZ9F9WMZ91BNQXR7	Duane Beevor	3	77
6	01HEDA9WSMFXW4ZN3W56YJE3V7	Kalie Chasles	11	76
7	01HEDA9WSNPWZVG2FV9PCGY5X7	Ruthe Revens	18	65
8	01HEDA9WSQZRBATG8DFFVEM4CB	Irena McNirlan	8	8
9	01HEDA9WSR0P4SCZ99KM72BJPR	Dolorita Thursby	5	76
10	01HEDA9WSSQXS20MP15GHTHCJN	Sophi Wennington	13	58
11	01HEDA9WSVP6Y25SXGX5Y4D1YK	Waldon McNamee	24	49
12	01HEDA9WSW8PFQF6BKKE1YHF1F	Cammy Gitsham	16	71
13	01HEDA9WSXE90NDVSBDF16QK7N	Sinclair Frushard	21	6

3. Retrieve the list of all materials along with their suppliers.	SELECT m.Material_ID, m.Name AS Material_Name, m.type AS Material_Type, s.Supplier_ID, s.Name AS Supplier_Name FROM MATERIAL m JOIN SUPPLIER s ON m.Supplier_ID = s.Supplier_ID;

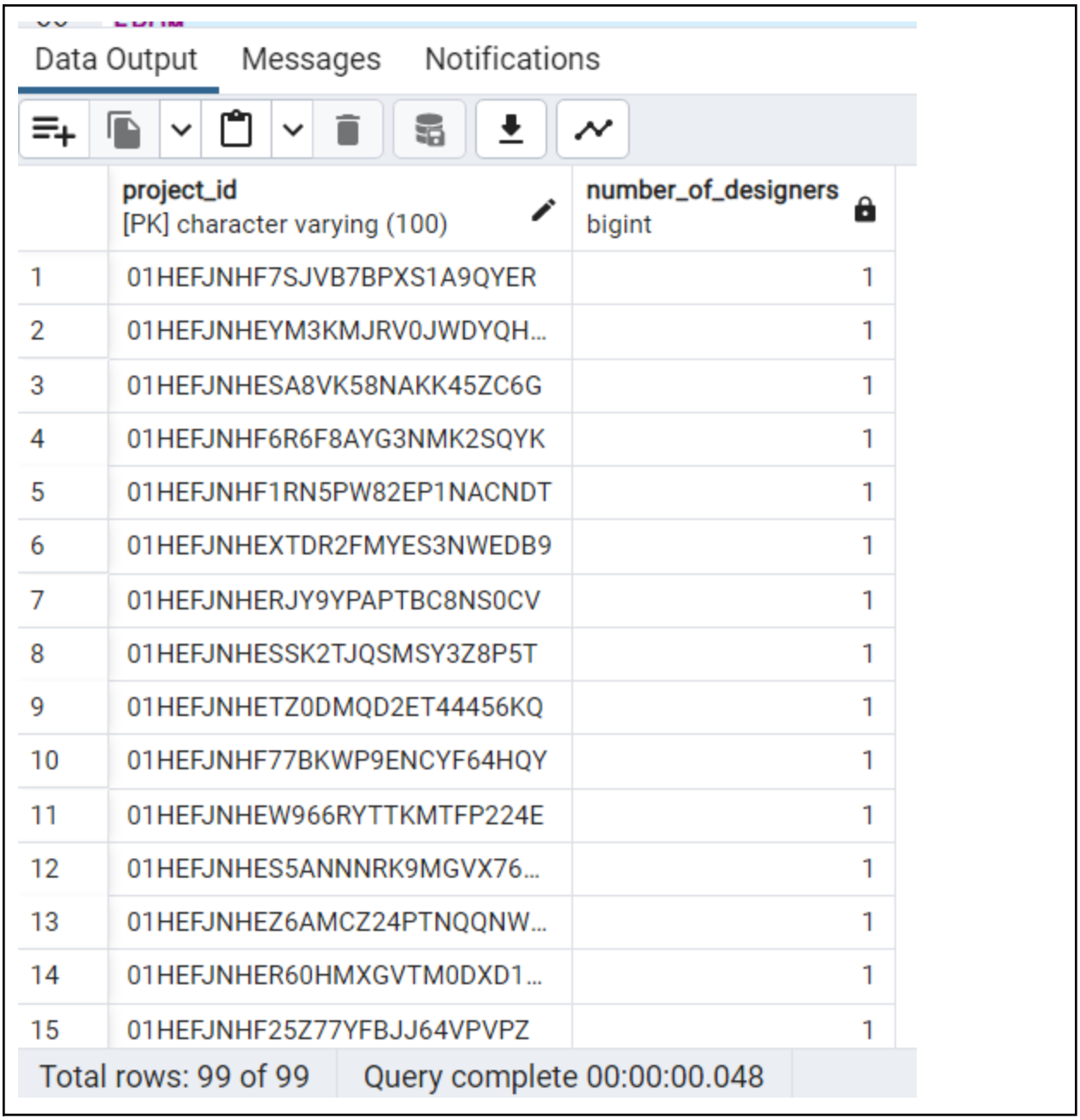
	material_id character varying (100)	material_name text	material_type text	supplier_id character varying (100)	supplier_name text
1	01HEDASZ6E0HW1QF7T42VT2CSE	Amelia Tinson	Vinyl	01HEDAH4BS7M8PTAG97TJKYXW7	Shaylynn Saiz
2	01HEDASZ6F0SP4S6ZBDFQ5M8SG	Roseline Whitman	Stone	01HEDAH4BTT1NG84RKMV8Y8670	Rodolphe Oldrey
3	01HEDASZ6FDVD5KVW3TN8ZFCPG	Juliette Gawke	Rubber	01HEDAH4BVH0KX5TD2JGB9YT2N	Heida Frewer
4	01HEDASZ6GCHC4W6BH7MGYT7…	Sutherlan Allchorn	Brass	01HEDAH4BWTV39ZX8T33FH1J01	Hyacinth Scowcroft
5	01HEDASZ6HC2E1EBVQV7KXF3FF	Teddie Cannaway	Wood	01HEDAH4BXMFTS70C4G85BJJM1	Iona Abbey
6	01HEDASZ6HA8T9BV6HJA55BSP1	Terra Richt	Plexiglass	01HEDAH4BXMFTS70C4G85BJJM1	Iona Abbey
7	01HEDASZ6J8TVEHN8WZSG51XK4	Bradly Simonnot	Vinyl	01HEDAH4BXMFTS70C4G85BJJM1	Iona Abbey
8	01HEDASZ6J6FB7TSTWQVW3P03J	Modestine Larmet	Granite	01HEDAH4BXMFTS70C4G85BJJM1	Iona Abbey
9	01HEDASZ6K0Q3710K2KSC8YW3G	Vevay Shailer	Brass	01HEDAH4BXMFTS70C4G85BJJM1	Iona Abbey
10	01HEDASZ6MH5PVMHM9NMPX5…	Guglielmo Flecknell	Aluminum	01HEDAH4BXMFTS70C4G85BJJM1	Iona Abbey
11	01HEDASZ6MCZPTQ6EPVX5BDEN4	Bertrand Poole	Plastic	01HEDAH4BXMFTS70C4G85BJJM1	Iona Abbey
12	01HEDASZ6NCH91HYDM95QB98T1	Leesa Husher	Vinyl	01HEDAH4BXMFTS70C4G85BJJM1	Iona Abbey
13	01HEDASZ6NS6C044GN4VRZ52NA	Christen Corrado	Plastic	01HEDAH4BXMFTS70C4G85BJJM1	Iona Abbey

4. Retrieve the list of all the projects along with the designers name working on them

```
SELECT
   p.Project_id,
   d.Name AS Designer_Name
FROM
   PROJECT p
JOIN
   DESIGNER d
   ON p.Designer_ID = d.Designer_ID;
```

	project_id character varying (100)	designer_name text
1	01HEFJNHERJY9YPAPTBC8NS0CV	Violet Grafhom
2	01HEFJNHER60HMXGVTM0DXD1P3	Gussie Sebert
3	01HEFJNHER9TPJZF38ZF7VZ7BA	Gustav Harkness
4	01HEFJNHER6VA94RGNHYJV5FV9	Jackelyn Petegrew
5	01HEFJNHESA8VK58NAKK45ZC6G	Duane Beevor
6	01HEFJNHESSK2TJQSMSY3Z8P5T	Kalie Chasles
7	01HEFJNHESCM3BRVZXFTT8MXJM	Ruthe Revens
8	01HEFJNHESZ5Y9CF8GBFYZMAQ7	Irena McNirlan
9	01HEFJNHES5ANNNRK9MGVX762Q	Dolorita Thursby
10	01HEFJNHETXD6RKVS3ZCGDR4N4	Sophi Wennington
11	01HEFJNHET0W3E3PGP45BHA2TA	Waldon McNamee
12	01HEFJNHETZ0DMQD2ET44456KQ	Cammy Gitsham

5. Count the number of designers per project.	SELECT p.Project_id, COUNT(d.Designer_ID) AS Number_of_Designers FROM PROJECT p LEFT JOIN DESIGNER d ON p.Designer_ID = d.Designer_ID GROUP BY p.Project_id;



Data Output | Messages | Notifications

	project_id [PK] character varying (100)	number_of_designers bigint
1	01HEFJNHF7SJVB7BPXS1A9QYER	1
2	01HEFJNHEYM3KMJRV0JWDYQH...	1
3	01HEFJNHESA8VK58NAKK45ZC6G	1
4	01HEFJNHF6R6F8AYG3NMK2SQYK	1
5	01HEFJNHF1RN5PW82EP1NACNDT	1
6	01HEFJNHEXTDR2FMYES3NWEDB9	1
7	01HEFJNHERJY9YPAPTBC8NS0CV	1
8	01HEFJNHESSK2TJQSMSY3Z8P5T	1
9	01HEFJNHETZ0DMQD2ET44456KQ	1
10	01HEFJNHF77BKWP9ENCYF64HQY	1
11	01HEFJNHEW966RYTTKMTFP224E	1
12	01HEFJNHES5ANNNRK9MGVX76...	1
13	01HEFJNHEZ6AMCZ24PTNQQNW...	1
14	01HEFJNHER60HMXGVTM0DXD1...	1
15	01HEFJNHF25Z77YFBJJ64VPVPZ	1

Total rows: 99 of 99 Query complete 00:00:00.048

6. Create a view that displays suppliers name and their phone number.	CREATE VIEW SupplierContactInfo AS SELECT Supplier_ID, Name AS Supplier_Name, Phone_no FROM SUPPLIER;

	project_id [PK] character varying (100)	number_of_designers bigint
1	01HEFJNHF7SJVB7BPXS1A9QYER	1
2	01HEFJNHEYM3KMJRV0JWDYQH...	1
3	01HEFJNHESA8VK58NAKK45ZC6G	1
4	01HEFJNHF6R6F8AYG3NMK2SQYK	1
5	01HEFJNHF1RN5PW82EP1NACNDT	1
6	01HEFJNHEXTDR2FMYES3NWEDB9	1
7	01HEFJNHERJY9YPAPTBC8NS0CV	1
8	01HEFJNHESSK2TJQSMSY3Z8P5T	1
9	01HEFJNHETZ0DMQD2ET44456KQ	1
10	01HEFJNHF77BKWP9ENCYF64HQY	1
11	01HEFJNHEW966RYTTKMTFP224E	1
12	01HEFJNHES5ANNNRK9MGVX76...	1

7. Create a view that displays the project id and their rating.	CREATE VIEW ProjectRating AS SELECT p.Project_id, cr.Rating FROM PROJECT p LEFT JOIN CLIENT_RATING cr ON p.Client_ID = cr.Client_ID; select * from projectrating;

	project_id character varying (100)	rating double precision
1	01HEFJNHERJY9YPAPTBC8NS0CV	4
2	01HEFJNHER60HMXGVTM0DXD1P3	3
3	01HEFJNHER9TPJZF38ZF7VZ7BA	5
4	01HEFJNHER6VA94RGNHYJV5FV9	5
5	01HEFJNHESA8VK58NAKK45ZC6G	5
6	01HEFJNHESSK2TJQSMSY3Z8P5T	2
7	01HEFJNHESCM3BRVZXFTT8MXJM	4
8	01HEFJNHESZ5Y9CF8GBFYZMAQ7	1
9	01HEFJNHES5ANNNRK9MGVX762Q	5
10	01HEFJNHETXD6RKVS3ZCGDR4N4	3
11	01HEFJNHET0W3E3PGP45BHA2TA	2

8. List all the projects by 2023-01-01.	SELECT project_id, end_date FROM project_details WHERE end_date < '2023-01-01';

	project_id character varying (100)	end_date date
1	01HEFJNHER60HMXGVTM0DXD1...	2022-08-08
2	01HEFJNHESCM3BRVZXFTT8MX...	2022-12-28
3	01HEFJNHESZ5Y9CF8GBFYZMA...	2022-10-02
4	01HEFJNHETXD6RKVS3ZCGDR4...	2022-05-08
5	01HEFJNHET0W3E3PGP45BHA2...	2022-04-15
6	01HEFJNHET4TJPYZ6DW0QDGS...	2022-10-25
7	01HEFJNHETJWX1MC4BRDYB3C...	2022-10-15
8	01HEFJNHEVBNVHT9M0JQE0VS...	2022-09-03
9	01HEFJNHEVDBSPC04E2TJ73NDV	2022-09-25
10	01HEFJNHEV8G0JXA4MJR6B9FAJ	2022-08-28
11	01HEFJNHEWZJBB4CGP9KNWVA...	2022-05-01
12	01HEFJNHEYM3KMJRV0JWDYQ...	2022-03-19

9. Retrieve the designers name who worked on a maximum number of projects.	SELECT d.Name AS Designer_Name FROM DESIGNER d WHERE (SELECT COUNT(*) FROM PROJECT p WHERE p.Designer_ID = d.Designer_ID) = (SELECT MAX(project_count) FROM (SELECT Designer_ID, COUNT(*) AS project_count FROM PROJECT GROUP BY Designer_ID) AS project_counts);

	designer_name text
1	Marne Worledge
2	Forest Puttick

10. List projects in ascending order of their start date.	SELECT Project_id, Start_Date FROM PROJECT_DETAILS ORDER BY Start_Date ASC;

	project_id character varying (100)	start_date date
1	01HEFJNHF07W4KT39RJJM5KJ0S	2019-01-01
2	01HEFJNHET0W3E3PGP45BHA2TA	2019-01-04
3	01HEFJNHF7M1VW5NC6YQPTT1SE	2019-01-19
4	01HEFJNHEVBNVHT9M0JQE0VSSK	2019-01-25
5	01HEFJNHFA102CVJ2KABYSZ28X	2019-03-02
6	01HEFJNHF4E7WNPWS5Z54FZB0K	2019-03-10
7	01HEFJNHF0QTDZ565HK1X0WCZM	2019-04-08
8	01HEFJNHF3JYAPWR5M7DYYMK...	2019-04-19
9	01HEFJNHETXD6RKVS3ZCGDR4N4	2019-05-01
10	01HEFJNHFADT0HSH2WH0XP7YX3	2019-05-08
11	01HEFJNHEZ4FW52MB5OS8S2A7B	2019-05-26

11. Create a view that displays the material type and their suppliers.	CREATE VIEW MaterialSupplier AS SELECT

	m.type AS Material_Type, m.Material_ID, m.Supplier_ID, s.Name AS Supplier_Name FROM MATERIAL m JOIN SUPPLIER s ON m.Supplier_ID = s.Supplier_ID; select * from materialsupplier;

	material_type text	material_id character varying (100)	supplier_id character varying (100)	supplier_name text
1	Vinyl	01HEDASZ6E0HW1QF7T42VT2CSE	01HEDAH4BS7M8PTAG97TJKYXW7	Shaylynn Saiz
2	Stone	01HEDASZ6F0SP4S6ZBDFQ5M8SG	01HEDAH4BTT1NG84RKMV8Y8670	Rodolphe Oldrey
3	Rubber	01HEDASZ6FDVD5KVW3TN8ZFCPG	01HEDAH4BVH0KX5TD2JGB9YT2N	Heida Frewer
4	Brass	01HEDASZ6GCHC4W6BH7MGYT7...	01HEDAH4BWTV39ZX8T33FH1J01	Hyacinth Scowcroft
5	Wood	01HEDASZ6HC2E1EBVQV7KXF3FF	01HEDAH4BXMFTS70C4G85BJJM1	Iona Abbey
6	Plexiglass	01HEDASZ6HA8T9BV6HJA55BSP1	01HEDAH4BXMFTS70C4G85BJJM1	Iona Abbey
7	Vinyl	01HEDASZ6J8TVEHN8WZSG51XK4	01HEDAH4BXMFTS70C4G85BJJM1	Iona Abbey
8	Granite	01HEDASZ6J6FB7TSTWQVW3P03J	01HEDAH4BXMFTS70C4G85BJJM1	Iona Abbey
9	Brass	01HEDASZ6K0Q3710K2KSC8YW3G	01HEDAH4BXMFTS70C4G85BJJM1	Iona Abbey
10	Aluminum	01HEDASZ6MH5PVMHM9NMPX5...	01HEDAH4BXMFTS70C4G85BJJM1	Iona Abbey
11	Plastic	01HEDASZ6MCZPTQ6EPVX5BDEN4	01HEDAH4BXMFTS70C4G85BJJM1	Iona Abbey
12	Vinyl	01HEDASZ6NCH91HYDM95QB98T1	01HEDAH4BXMFTS70C4G85BJJM1	Iona Abbey
13	Plastic	01HEDASZ6NS6C044GN4VRZ52NA	01HEDAH4BXMFTS70C4G85BJJM1	Iona Abbey

12. List all the transactions greater than $650.	SELECT transaction_id, amount FROM transactions WHERE amount > '$650';

Data Output | Messages | Notifications

	transaction_id [PK] character varying (100)	amount character varying (100)
1	01HEDCEHB47Q8PSX51XT23VEZE	$935.57
2	01HEDCEHB5DFT8E6T9XVTHRCPE	$963.89
3	01HEDCEHB7WC291FFSV2TDE6DS	$847.15
4	01HEDCEHB8Q7KQGQH8YTM3SZ...	$867.37
5	01HEDCEHB9TPDN9SEZZKJ79BKH	$834.23
6	01HEDCEHBA60KY937G93BDWN...	$755.48
7	01HEDCEHBBABC7W2P8AJACBB...	$691.82
8	01HEDCEHBCGKHM7MKTYN5JY7...	$789.41
9	01HEDCEHBDGVM082R1ETCV0V...	$781.36
10	01HEDCEHBHBPWX26J19ANG8F...	$824.45
11	01HEDCEHBP4K9KF91Z6YBJHX73	$725.21
12	01HEDCEHBQGYT5Y9GYYE5TXSBB	$826.41
13	01HEDCEHBS1PP1PC0SP3A7XZ9Q	$792.26
14	01HEDCEHBV4VT1V9A3DD4X4TXR	$919.47
15	01HEDCEHBWRT5SMR4C2PBHPY...	$938.17

Total rows: 69 of 69 | Query complete 00:00:00.047

13. Fetch all the clients with their respective ratings:	SELECT CLIENT.Client_ID, CLIENT.Name, CLIENT_RATING.Rating FROM CLIENT INNER JOIN CLIENT_RATING ON CLIENT.Client_ID = CLIENT_RATING.Client_ID;

	transaction_id [PK] character varying (100)	amount character varying (100)
1	01HEDCEHB47Q8PSX51XT23VEZE	$935.57
2	01HEDCEHB5DFT8E6T9XVTHRCPE	$963.89
3	01HEDCEHB7WC291FFSV2TDE6DS	$847.15
4	01HEDCEHB8Q7KQGQH8YTM3SZ...	$867.37
5	01HEDCEHB9TPDN9SEZZKJ79BKH	$834.23
6	01HEDCEHBA60KY937G93BDWN...	$755.48
7	01HEDCEHBBABC7W2P8AJACBB...	$691.82
8	01HEDCEHBCGKHM7MKTYN5JY7...	$789.41
9	01HEDCEHBDGVM082R1ETCV0V...	$781.36
10	01HEDCEHBHBPWX26J19ANG8F...	$824.45
11	01HEDCEHBP4K9KF91Z6YBJHX73	$725.21
12	01HEDCEHBQGYT5Y9GYYE5TXSBB	$826.41

14. Fetch the details of the project with the longest duration:	SELECT PROJECT.Project_id, (PROJECT_DETAILS.End_Date - PROJECT_DETAILS.Start_Date) as Duration_in_days FROM PROJECT INNER JOIN PROJECT_DETAILS ON PROJECT.Project_id = PROJECT_DETAILS.Project_id ORDER BY Duration_in_days DESC LIMIT 1;

	project_id [PK] character varying (100)	duration_in_days integer
1	01HEFJNHF4E7WNPWS5Z54FZB0K	1676

15. Fetch the details of the supplier providing the most number of materials:	SELECT material.Supplier_ID, material.type, COUNT(Material_ID) as Number_of_Materials FROM MATERIAL GROUP BY MATERIAL.Supplier_ID, material.type

	ORDER BY Number_of_Materials DESC LIMIT 1;

	supplier_id character varying (100)	type text	number_of_materials bigint
1	01HEDAH4C8XBYK9M1NYEAGM5CK	Wood	3

16. Fetch the details of the most expensive material:	SELECT MATERIAL.Material_ID, MATERIAL.type, MATERIAL_COST.Price FROM MATERIAL INNER JOIN MATERIAL_COST ON MATERIAL.Material_ID = MATERIAL_COST.Material_ID ORDER BY MATERIAL_COST.Price DESC LIMIT 1;

	material_id character varying (100)	type text	price text
1	01HEDASZ92F5N0GSEQDD4CHBCZ	Brass	$99.61

17. Fetch the details of the admin who has supervised the least projects:	SELECT Admin_ID, COUNT(Project_id) as Number_of_Projects FROM PROJECT GROUP BY Admin_ID ORDER BY Number_of_Projects ASC LIMIT 1;

	admin_id [PK] character varying (100)	admin_name text	number_of_projects bigint
1	01HEDC2T24EPVJT2GW4PD2GQZX	Ruby James	1

18. Fetch the details of the admin who has supervised the most projects and also fetch the admin name:	SELECT A.Admin_ID, A.Name AS Admin_Name, P.Number_of_Projects FROM ADMINS A JOIN (

	SELECT Admin_ID, COUNT(Project_id) as Number_of_Projects FROM PROJECT GROUP BY Admin_ID) P ON A.Admin_ID = P.Admin_ID ORDER BY Number_of_Projects DESC LIMIT 1;

	admin_id [PK] character varying (100)	admin_name text	number_of_projects bigint
1	01HEDC2T22MSFX1708YB0ZSW8K	Mathew perris	36

19. Find Architects with Highest Experience:	SELECT A.Architect_id, A.Name AS Architect_Name, A.Years_of_Experience FROM Architect A WHERE A.Years_of_Experience = (SELECT MAX(Years_of_Experience) FROM Architect);

	architect_id [PK] character varying (100)	architect_name text	years_of_experience bigint
1	01HEFEZ91HNTE10GAVYN9EMWP8	Flora Halgarth	20
2	01HEFEZ928D9WH2XGJG7QCG7NP	Osbourn Kubes	20

20. List Suppliers and Their Materials along with Material Quantities and also create a view for the same and load results:	CREATE VIEW SupplierMaterialQuantity AS SELECT S.Supplier_ID, S.Name AS Supplier_Name, M.Material_ID, M.Name AS Material_Name, MQ.Quantity FROM SUPPLIER S JOIN MATERIAL M ON S.Supplier_ID = M.Supplier_ID JOIN MATERIAL_QUANTITY MQ ON M.Material_ID = MQ.Material_ID AND M.Supplier_ID = MQ.Supplier_ID; select * from suppliermaterialquantity;

	supplier_id character varying (100)	supplier_name text	material_id character varying (100)	material_name text	quantity bigint
1	01HEDAH4BS7M8PTAG97TJKYXW7	Shaylynn Saiz	01HEDASZ6E0HW1QF7T42VT2CSE	Amelia Tinson	41
2	01HEDAH4BTT1NG84RKMV8Y8670	Rodolphe Oldrey	01HEDASZ6F0SP4S6ZBDFQ5M8SG	Roseline Whitman	61
3	01HEDAH4BVH0KX5TD2JGB9YT2N	Heida Frewer	01HEDASZ6FDVD5KVW3TN8ZFCPG	Juliette Gawke	58
4	01HEDAH4BWTV39ZX8T33FH1J01	Hyacinth Scowcroft	01HEDASZ6GCHC4W6BH7MGYT7...	Sutherlan Allchorn	56
5	01HEDAH4BXMFTS70C4G85BJJM1	Iona Abbey	01HEDASZ6HC2E1EBVQV7KXF3FF	Teddie Cannaway	71
6	01HEDAH4BXMFTS70C4G85BJJM1	Iona Abbey	01HEDASZ6HA8T9BV6HJA55BSP1	Terra Richt	41
7	01HEDAH4BXMFTS70C4G85BJJM1	Iona Abbey	01HEDASZ6J8TVEHN8WZSG51XK4	Bradly Simonnot	10
8	01HEDAH4BXMFTS70C4G85BJJM1	Iona Abbey	01HEDASZ6J6FB7TSTWQVW3P03J	Modestine Larmet	39
9	01HEDAH4BXMFTS70C4G85BJJM1	Iona Abbey	01HEDASZ6K0Q3710K2KSC8YW3G	Vevay Shailer	49
10	01HEDAH4BXMFTS70C4G85BJJM1	Iona Abbey	01HEDASZ6MH5PVMHM9NMPX5...	Guglielmo Flecknell	22
11	01HEDAH4BXMFTS70C4G85BJJM1	Iona Abbey	01HEDASZ6MCZPTQ6EPVX5BDEN4	Bertrand Poole	62
12	01HEDAH4BXMFTS70C4G85BJJM1	Iona Abbey	01HEDASZ6NCH91HYDM95QB98T1	Leesa Husher	22
13	01HEDAH4BXMFTS70C4G85BJJM1	Iona Abbey	01HEDASZ6NS6C044GN4VRZ52NA	Christen Corrado	51

21. List Projects with Designers Having More Projects Than Average:	SELECT P.Project_id, P.Client_ID, P.Designer_ID
	FROM PROJECT P
	WHERE P.Designer_ID IN (
	  SELECT Designer_ID
	  FROM DESIGNER
	  WHERE No_of_Projects > (
	  SELECT AVG(No_of_Projects)
	  FROM DESIGNER
	  )
	);

	project_id [PK] character varying (100)	client_id character varying (100)	designer_id character varying (100)
1	01HEFJNHERJY9YPAPTBC8NS0CV	01HED9J0ZN8TA6XXQVY0CBZ72J	01HEDA9WSDH8A9C51AD91R29DF
2	01HEFJNHESA8VK58NAKK45ZC6G	01HED9J0ZT6DFXENP2Q6K7J8M6	01HEDA9WSKQZ9F9WMZ91BNQXR7
3	01HEFJNHESSK2TJQSMSY3Z8P5T	01HED9J0ZTJGPEZ0KF81T3RHEB	01HEDA9WSMFXW4ZN3W56YJE3V7
4	01HEFJNHESCM3BRVZXFTT8MX...	01HED9J0ZV4SXRYQR25MCPZ1...	01HEDA9WSNPWZVG2FV9PCGY5X7
5	01HEFJNHES5ANNNRK9MGVX76...	01HED9J0ZY56VWZMNDV287M...	01HEDA9WSR0P4SCZ99KM72BJPR
6	01HEFJNHETXD6RKVS3ZCGDR4...	01HED9J0ZZGZB2M5S0QFXB064R	01HEDA9WSSQXS20MP15GHTHCJN
7	01HEFJNHET0W3E3PGP45BHA2...	01HED9J100QXG5QZN8ND1QY2...	01HEDA9WSVP6Y25SXGX5Y4D1YK
8	01HEFJNHETZ0DMQD2ET44456K...	01HED9J1006F663F3P34M2YNK9	01HEDA9WSW8PFQF6BKKE1YHF1F
9	01HEFJNHET4B7GWDZYNT595F30	01HED9J103VZQ9RMYGVM4Y47...	01HEDA9WSYF0B776C0AA4RFCZV
10	01HEFJNHEVX3S8FGBVNCMMSP...	01HED9J105DF872C84Y3PXQ1X6	01HEDA9WT2TY1Y48APE4B85SZK
11	01HEFJNHEVDBSPC04E2TJ73NDV	01HED9J107HDT25J123HNWYZ...	01HEDA9WT53HD0M27ECWHW52TC
12	01HEFJNHEV8G0JXA4MJR6B9FAJ	01HED9J108M8M36VVFVKMXQ1...	01HEDA9WT6E94C1Y87QR2W433B
13	01HEFJNHEW966RYTTKMTFP22...	01HED9J109AW8QMCFFSSVH7P...	01HEDA9WT87XRERAM3DQJ6JY5N

22. Calculate Average Material Quantity for Each Material:	SELECT M.Material_ID, AVG(MQ.Quantity) AS Avg_Quantity FROM MATERIAL M JOIN MATERIAL_QUANTITY MQ ON M.Material_ID = MQ.Material_ID AND M.Supplier_ID = MQ.Supplier_ID GROUP BY M.Material_ID;

	material_id character varying (100)	avg_quantity numeric
1	01HEDASZ8FHES4F90WVP2NCS40	42.0000000000000000
2	01HEDASZ9DA03B32MWMF4KV2...	48.0000000000000000
3	01HEDASZ8GK1VVASJ75FP6YVAA	12.0000000000000000
4	01HEDASZ6PJPKE6WQ6GNSJY7E...	53.0000000000000000
5	01HEDASZ9N68N9V0S14YMHCS...	79.0000000000000000
6	01HEDASZ8HPPD0YZG7QASNTQ...	49.0000000000000000
7	01HEDASZ8YCWQ04APPWXT9Z6...	21.0000000000000000
8	01HEDASZ6E0HW1QF7T42VT2CSE	41.0000000000000000
9	01HEDASZ9G6HMRBNDHF7XDK9...	44.0000000000000000
10	01HEDASZ88WW4184WC1X5V05...	42.0000000000000000
11	01HEDASZ9KRS01PPT7B071HP79	62.0000000000000000
12	01HEDASZ924SCTWSNYXCQN23...	43.0000000000000000

23. Fetch details of project cost with	SELECT P.Project_id, M.type AS Material_type, MC.Price AS Material_Cost FROM PROJECT P JOIN MATERIAL M ON P.Material_ID = M.Material_ID AND P.Supplier_ID = M.Supplier_ID JOIN MATERIAL_COST MC ON P.Material_ID = MC.Material_ID AND P.Supplier_ID = MC.Supplier_ID;

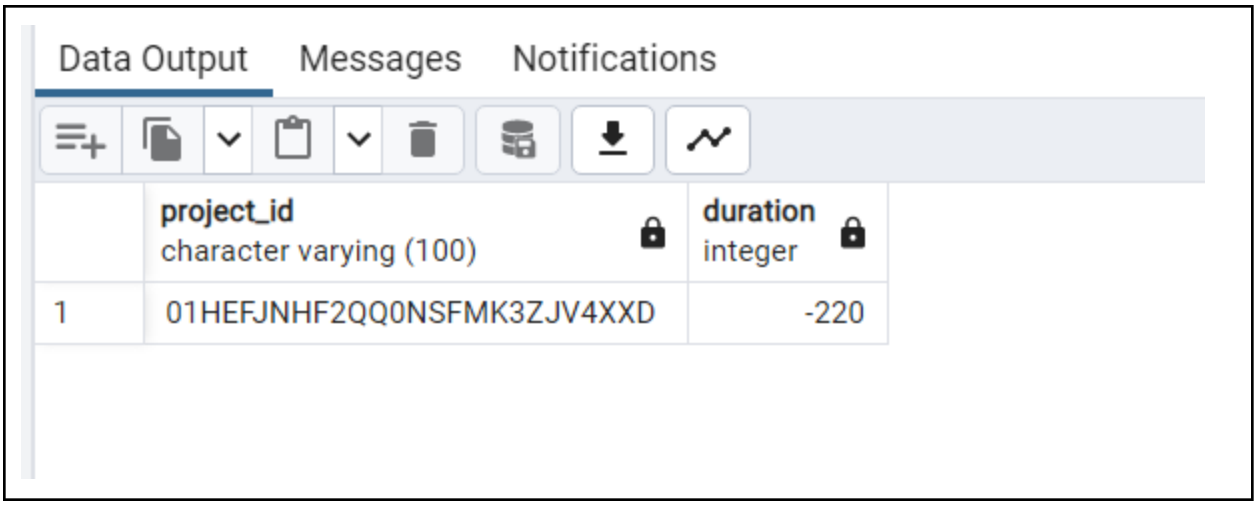
	project_id character varying (100)	material_type text	material_cost text
1	01HEFJNHERJY9YPAPTBC8NS0CV	Vinyl	$137.36
2	01HEFJNHER60HMXGVTM0DXD1P3	Stone	$59.68
3	01HEFJNHER9TPJZF38ZF7VZ7BA	Rubber	$170.42
4	01HEFJNHER6VA94RGNHYJV5FV9	Brass	$196.39
5	01HEFJNHESA8VK58NAKK45ZC6G	Wood	$99.26
6	01HEFJNHESSK2TJQSMSY3Z8P5T	Plexiglass	$166.78
7	01HEFJNHESCM3BRVZXFTT8MXJM	Vinyl	$33.44
8	01HEFJNHESZ5Y9CF8GBFYZMAQ7	Granite	$68.16
9	01HEFJNHES5ANNNRK9MGVX762Q	Brass	$138.37
10	01HEFJNHETXD6RKVS3ZCGDR4N4	Aluminum	$123.86
11	01HEFJNHET0W3E3PGP45BHA2TA	Plastic	$147.61
12	01HEFJNHETZ0DMQD2ET44456KQ	Vinyl	$172.94

24. List all ongoing projects.	SELECT Project_id, Start_Date, End_Date FROM PROJECT_DETAILS WHERE Start_Date <= CURRENT_DATE AND End_Date >= CURRENT_DATE;

Data Output | Messages | Notifications

	project_id character varying (100)	start_date date	end_date date
1	01HEFJNHESSK2TJQSMSY3Z8P5T	2020-09-12	2023-12-29
2	01HEFJNHEVSA851B4AT2R12K9Z	2023-01-16	2024-01-07
3	01HEFJNHEZZZ34BF26NZ9NBPAP	2021-05-27	2024-03-10
4	01HEFJNHEZF3BF0K5B1JY7EVW2	2019-07-14	2024-01-04
5	01HEFJNHF0YWK754651BF10JG8	2023-01-26	2023-11-27
6	01HEFJNHF2A4RKAA8ADPBDHG3Q	2022-11-30	2024-02-11
7	01HEFJNHF4DR54HK5HJQPDX3XH	2023-10-17	2024-03-24
8	01HEFJNHF645RQSSJJ33NAXK0Q	2021-01-25	2024-04-15
9	01HEFJNHF6JBBNVR15J2W6DYTG	2021-01-03	2024-04-04
10	01HEFJNHF7YCQ17YMJGK1CQ93R	2019-10-16	2024-03-22
11	01HEFJNHF8SBE4PA6P6QEWSNSY	2021-11-02	2024-03-13
12	01HEFJNHF802TGJ64Z5Y40D467	2022-12-10	2024-03-23
13	01HEFJNHF9Y1JSX6TRRDEAGW3P	2019-12-11	2023-12-09
14	01HEFJNHF9HAQW0YZ3AZC60CRN	2022-06-20	2024-03-11
15	01HEFJNHFBRH14SERY3XXWV7TP	2021-06-08	2024-02-20
16	01HEFJNHFBQE7WMGKYWYRG9E8G	2020-08-09	2024-03-25

25. Which project took the least amount of time.	SELECT Project_id, (End_Date - Start_Date) AS Duration FROM PROJECT_DETAILS ORDER BY Duration LIMIT 1;



26. List of all the materials.	SELECT Material_ID, Name, Type FROM MATERIAL;

Data Output | Messages | Notifications

	material_id character varying (100)	name text	type text
1	01HEDASZ6E0HW1QF7T42VT2CSE	Amelia Tinson	Vinyl
2	01HEDASZ6F0SP4S6ZBDFQ5M8SG	Roseline Whitman	Stone
3	01HEDASZ6FDVD5KVW3TN8ZFCPG	Juliette Gawke	Rubber
4	01HEDASZ6GCHC4W6BH7MGYT7...	Sutherlan Allchorn	Brass
5	01HEDASZ6HC2E1EBVQV7KXF3FF	Teddie Cannaway	Wood
6	01HEDASZ6HA8T9BV6HJA55BSP1	Terra Richt	Plexiglass
7	01HEDASZ6J8TVEHN8WZSG51XK4	Bradly Simonnot	Vinyl
8	01HEDASZ6J6FB7TSTWQVW3P03J	Modestine Larmet	Granite
9	01HEDASZ6K0Q3710K2KSC8YW3G	Vevay Shailer	Brass
10	01HEDASZ6MH5PVMHM9NMPX5...	Guglielmo Flecknell	Aluminum
11	01HEDASZ6MCZPTQ6EPVX5BDEN4	Bertrand Poole	Plastic
12	01HEDASZ6NCH91HYDM95QB98T1	Leesa Husher	Vinyl
13	01HEDASZ6NS6C044GN4VRZ52NA	Christen Corrado	Plastic
14	01HEDASZ6PJPKE6WQ6GNSJY7EW	Steffen Danslow	Glass
15	01HEDASZ82FS39BBXZPEQ780N0	Liesa Manifold	Stone
16	01HEDASZ8283F87H43C83KG4Q9	Todd Deetlof	Wood
17	01HEDASZ830ZFNKYTS8BDEK4PA	Kile Myner	Vinyl
18	01HEDASZ84RSDAK47YE1TA5TR6	Jervis Buzek	Plastic
19	01HEDASZ850J1BVHMS87H7FNGR	Tobit Clink	Plastic
20	01HEDASZ856XZPM8WX7ERSYTNA	Filberto Lacase	Glass
21	01HEDASZ86DY1CAS3F8FQ9038X	Lorianne Waryk	Granite
22	01HEDASZ869FNFCNSHCZZ0NP2V	Lamar Ambrosin	Wood

Total rows: 100 of 100 | Query complete 00:00:00.062

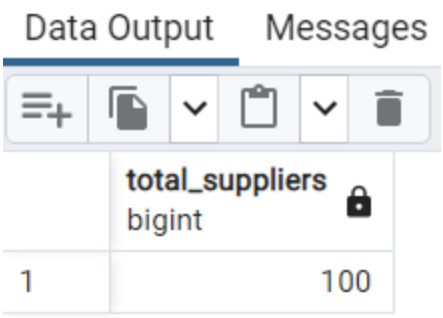
27. List of all the clients and their names.	SELECT client_id, name FROM CLIENT;

Data Output | Messages | Notifications

	client_id [PK] character varying (100)	name text
1	01HED9J0ZN8TA6XXQVY0CBZ72J	Raphaela Henstone
2	01HED9J0ZQW2MWPJMD25WP11...	Giustino Alpin
3	01HED9J0ZRNJVRQGHMNJ06EJMC	Stanley Hanley
4	01HED9J0ZSYBPEJD53J51WK5XM	Bridget Nevitt
5	01HED9J0ZT6DFXENP2Q6K7J8M6	Hermione Trattles
6	01HED9J0ZTJGPEZ0KF81T3RHEB	Hubey Quarless
7	01HED9J0ZV4SXRYQR25MCPZ1HK	Harri Rubra
8	01HED9J0ZW0VD2H3SD2GB9H4FG	Kerri Aire
9	01HED9J0ZY56VWZMNDV287MV3A	Jilleen Beynon
10	01HED9J0ZZGZB2M5S0QFXB064R	Simone Tack
11	01HED9J100QXG5QZN8ND1QY2CH	Nikolaos Biasi
12	01HED9J1006F663F3P34M2YNK9	Kerwin Jamison
13	01HED9J102W0355Y3N17FMEFZP	Deni Birdsey
14	01HED9J103VZQ9RMYGVM4Y47NT	Karola Schafer
15	01HED9J104016GFXFSZR6K1SBX	Lorene Paolazzi
16	01HED9J104ZQJCVS2W975W56QA	Kat Bounds
17	01HED9J105DF872C84Y3PXQ1X6	Josey Gaddie
18	01HED9J1065QR9MM83N5FSZRSQ	Lelah Fendlen
19	01HED9J107HDT25J123HNWYZP6	Hart Tilt
20	01HED9J108M8M36VVFVKMXQ1R5	Aggi Carverhill
21	01HED9J109AW8QMCFFSSVH7PA1	Violette Edmand
22	01HED9J10ANS8R4Q0CS4E4AXX4	Filberte Letourtre

Total rows: 100 of 100 | Query complete 00:00:00.046

28. Calculate the total number of suppliers.	SELECT COUNT(*) AS total_suppliers FROM SUPPLIER;



29. Calculate the number of materials used in each project based on type.	SELECT p.Project_id, m.type, SUM(mq.Quantity) AS total_quantity FROM PROJECT p JOIN MATERIAL m ON p.Material_ID = m.Material_ID JOIN MATERIAL_QUANTITY mq ON m.Material_ID = mq.Material_ID GROUP BY p.Project_id, m.type ORDER BY p.Project_id, m.type;

Data Output | Messages | Notifications

	project_id character varying (100)	type text	total_quantity numeric
1	01HEFJNHER60HMXGVTM0DXD1...	Stone	61
2	01HEFJNHER6VA94RGNHYJV5FV9	Brass	56
3	01HEFJNHER9TPJZF38ZF7VZ7BA	Rubber	58
4	01HEFJNHERJY9YPAPTBC8NS0CV	Vinyl	41
5	01HEFJNHES5ANNNRK9MGVX76...	Brass	49
6	01HEFJNHESA8VK58NAKK45ZC6G	Wood	71
7	01HEFJNHESCM3BRVZXFTT8MX...	Vinyl	10
8	01HEFJNHESSK2TJQSMSY3Z8P5T	Plexiglass	41
9	01HEFJNHESZ5Y9CF8GBFYZMA...	Granite	39
10	01HEFJNHET0W3E3PGP45BHA2...	Plastic	62
11	01HEFJNHET4B7GWDZYNT595F30	Glass	53
12	01HEFJNHET4TJPYZ6DW0QDGS...	Plastic	51
13	01HEFJNHETJWX1MC4BRDYB3C...	Stone	23
14	01HEFJNHETXD6RKVS3ZCGDR4...	Aluminum	22
15	01HEFJNHETZ0DMQD2ET44456K...	Vinyl	22
16	01HEFJNHEV8G0JXA4MJR6B9FAJ	Glass	46
17	01HEFJNHEVBNVHT9M0JQE0VS...	Plastic	32
18	01HEFJNHEVDBSPC04E2TJ73NDV	Plastic	71
19	01HEFJNHEVSA851B4AT2R12K9Z	Wood	48
20	01HEFJNHEVX3S8FGBVNCMMSP...	Vinyl	39
21	01HEFJNHEW966RYTTKMTFP22...	Granite	43

Total rows: 99 of 99 | Query complete 00:00:00.057

30. Retrieve the list of all the transactions along with the invoice number.	SELECT Transaction_ID, Invoice_No FROM TRANSACTIONS;

Data Output | Messages | Notifications

	transaction_id [PK] character varying (100)	invoice_no character varying (100)
1	01HEDCEHB47Q8PSX51XT23VEZE	443492986-0
2	01HEDCEHB5DFT8E6T9XVTHRCPE	791323629-9
3	01HEDCEHB7WC291FFSV2TDE6DS	750095959-1
4	01HEDCEHB8Q7KQGQH8YTM3SZ...	377764190-1
5	01HEDCEHB9TPDN9SEZZKJ79BKH	937852548-2
6	01HEDCEHBA60KY937G93BDWN...	856901631-X
7	01HEDCEHBBABC7W2P8AJACBB...	573933623-6
8	01HEDCEHBCGKHM7MKTYN5JY7...	803454588-0
9	01HEDCEHBDGVM082R1ETCV0V...	553842770-6
10	01HEDCEHBE4HSFV0YC0TJ2768M	247754265-6
11	01HEDCEHBF967XDSXMVE29FD6V	856863596-2
12	01HEDCEHBG56RA47G6XQZH8R1T	014898682-X
13	01HEDCEHBHBPWX26J19ANG8F...	020190840-9
14	01HEDCEHBJ8PX1MG66HXV4P3...	536301320-7
15	01HEDCEHBMES6QJF01SN1PBP94	961195045-4
16	01HEDCEHBNP0N4JA1TVQYTRGE7	671018675-X
17	01HEDCEHBNN74F4BF8ESRC088F	546655287-9
18	01HEDCEHBP4K9KF91Z6YBJHX73	279686978-4
19	01HEDCEHBQGYT5Y9GYYE5TXSBB	178287791-6
20	01HEDCEHBRT39VPFJ5420D0X6D	706514464-X
21	01HEDCEHBS1PP1PC0SP3A7XZ9Q	008429608-9
22	01HEDCEHBTFCCKD79DEQCVE0	857841071-8

Total rows: 100 of 100 | Query complete 00:00:00.049

31. Create a stored procedure for adding a new project to the interior designing database.	CREATE OR REPLACE PROCEDURE AddNewProject(
	  IN p_Project_id VARCHAR(100),
	  IN p_Client_ID VARCHAR(100),
	  IN p_Designer_ID VARCHAR(100),
	  IN p_Architect_ID VARCHAR(100),
	  IN p_Material_ID VARCHAR(100),
	  IN p_Admin_ID VARCHAR(100),
	  IN p_User_ID VARCHAR(100),
	  IN p_Start_Date DATE,
	  IN p_End_Date DATE,
	  IN p_Status VARCHAR(100)
	)
	AS $$
	BEGIN
	  -- Insert data into PROJECT table
	  INSERT INTO PROJECT(Project_id, Client_ID, Designer_ID, Architect_ID, Material_ID, Admin_ID, User_ID)
	  VALUES (p_Project_id, p_Client_ID, p_Designer_ID, p_Architect_ID, p_Material_ID, p_Admin_ID, p_User_ID);
	
	  -- Insert data into PROJECT_DETAILS table
	  INSERT INTO PROJECT_DETAILS(Project_id, Start_Date, End_Date, Status)
	  VALUES (p_Project_id, p_Start_Date, p_End_Date, p_Status);
	END;
	$$ LANGUAGE plpgsql;

```
CREATE PROCEDURE

Query returned successfully in 150 msec.
```

32. Find the material supplied by each supplier along with its price and quantity?	SELECT S.Name AS Supplier_Name, M.Name AS Material_Name, MC.Price, MQ.Quantity FROM SUPPLIER S JOIN MATERIAL_COST MC ON S.Supplier_ID = MC.Supplier_ID JOIN MATERIAL M ON MC.Material_ID = M.Material_ID JOIN MATERIAL_QUANTITY MQ ON M.Material_ID = MQ.Material_ID AND S.Supplier_ID = MQ.Supplier_ID;

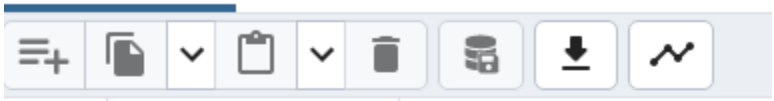
Data Output | Messages | Notifications

	supplier_name text	material_name text	price text	quantity bigint
1	Shaylynn Saiz	Amelia Tinson	$137.36	41
2	Rodolphe Oldrey	Roseline Whitman	$59.68	61
3	Heida Frewer	Juliette Gawke	$170.42	58
4	Hyacinth Scowcroft	Sutherlan Allchorn	$196.39	56
5	Iona Abbey	Teddie Cannaway	$99.26	71
6	Aviva Suthren	Terra Richt	$166.78	41
7	Hildegaard Castleman	Bradly Simonnot	$33.44	10
8	Evaleen Castellet	Modestine Larmet	$68.16	39
9	Leonhard Pottle	Vevay Shailer	$138.37	49
10	Simon Willisch	Guglielmo Flecknell	$123.86	22
11	Flin Burrows	Bertrand Poole	$147.61	62
12	Robenia Bassham	Leesa Husher	$172.94	22
13	Cathrine Scarratt	Christen Corrado	$44.22	51
14	Amitie Mordan	Steffen Danslow	$112.92	53
15	Anatola Jostan	Liesa Manifold	$129.51	23
16	Cristi Paternoster	Todd Deetlof	$124.24	48
17	Sylvester Benallack	Kile Myner	$119.66	39
18	Margit Fall	Jervis Buzek	$117.66	32
19	Jerome Halse	Tobit Clink	$83.41	71
20	Esme Barnish	Filberto Lacase	$28.78	46
21	Madella Sives	Lorianne Waryk	$73.12	43
22	Geoffrey Dallander	Lamar Ambrosin	$101.23	26

Total rows: 100 of 100 | Query complete 00:00:00.039

33. List the designers who have not worked on any project.	SELECT D.Name AS Designer_Name FROM DESIGNER D LEFT JOIN PROJECT P ON D.Designer_ID = P.Designer_ID WHERE P.Project_id IS NULL;

Data Output Messages Notifications



	designer_name text
1	Alberik Gillman
2	Tandy Rebichon
3	Neill Smiz
4	Sinclair Frushard
5	Georgi Hughlock
6	Crystie Kliner

34. Fetch all the projects that ended before '2023-01-01'.	SELECT project_id, start_date, end_date FROM project_details WHERE end_date < '2023-01-01';

Data Output | Messages | Notifications

	project_id character varying (100)	start_date date	end_date date
1	01HEFJNHER60HMXGVTM0DXD1...	2020-04-04	2022-08-08
2	01HEFJNHESCM3BRVZXFTT8MX...	2022-10-18	2022-12-28
3	01HEFJNHESZ5Y9CF8GBFYZMA...	2022-09-09	2022-10-02
4	01HEFJNHETXD6RKVS3ZCGDR4...	2019-05-01	2022-05-08
5	01HEFJNHET0W3E3PGP45BHA2...	2019-01-04	2022-04-15
6	01HEFJNHET4TJPYZ6DW0QDGS...	2021-05-12	2022-10-25
7	01HEFJNHETJWX1MC4BRDYB3C...	2021-09-10	2022-10-15
8	01HEFJNHEVBNVHT9M0JQE0VS...	2019-01-25	2022-09-03
9	01HEFJNHEVDBSPC04E2TJ73NDV	2021-10-15	2022-09-25
10	01HEFJNHEV8G0JXA4MJR6B9FAJ	2019-09-07	2022-08-28
11	01HEFJNHEWZJBB4CGP9KNWVA...	2021-04-26	2022-05-01
12	01HEFJNHEYM3KMJRV0JWDYQ...	2022-01-15	2022-03-19
13	01HEFJNHEY62JN46Y1S2X2782A	2021-08-06	2022-11-20
14	01HEFJNHEY90RSTRWWBRVTAF...	2020-06-07	2022-10-07
15	01HEFJNHF07910F1GVJJ2KWK5D	2022-10-19	2022-09-10
16	01HEFJNHF0984J1AG6P841G29V	2020-04-13	2022-10-13
17	01HEFJNHF1RN5PW82EP1NACN...	2022-04-21	2022-05-23
18	01HEFJNHF18396WW2N2SEGCN...	2019-08-06	2022-09-27
19	01HEFJNHF1AP9PQTJXX5DXRJPJ	2021-07-19	2022-08-31
20	01HEFJNHF2ACKMMEE7E7X50P...	2020-05-29	2022-07-20
21	01HEFJNHF3EHJGSMFYRHRGDN...	2022-11-02	2022-07-08

Total rows: 33 of 33 | Query complete 00:00:00.046

35. Fetch projects by ascending order of start_date	SELECT project_id, start_date FROM project_details ORDER BY start_date ASC;

Data Output Messages Notifications

	project_id character varying (100)	start_date date
1	01HEFJNHF07W4KT39RJJM5KJ0S	2019-01-01
2	01HEFJNHET0W3E3PGP45BHA2TA	2019-01-04
3	01HEFJNHF7M1VW5NC6YQPTT1SE	2019-01-19
4	01HEFJNHEVBNVHT9M0JQE0VSSK	2019-01-25
5	01HEFJNHFA102CVJ2KABYSZ28X	2019-03-02
6	01HEFJNHF4E7WNPWS5Z54FZB0K	2019-03-10
7	01HEFJNHF0QTDZ565HK1X0WCZM	2019-04-08
8	01HEFJNHF3JYAPWR5M7DYYMK...	2019-04-19
9	01HEFJNHETXD6RKVS3ZCGDR4N4	2019-05-01
10	01HEFJNHFADT0HSH2WH0XP7YX3	2019-05-08
11	01HEFJNHEZ4FW52MB5QS8S2A7B	2019-05-26
12	01HEFJNHF4KF4EWQ6N5J1TRCNV	2019-06-07
13	01HEFJNHEZF3BF0K5B1JY7EVW2	2019-07-14
14	01HEFJNHEX668Z32FVMHKYYCN4	2019-07-21
15	01HEFJNHF18396WW2N2SEGCNZ...	2019-08-06
16	01HEFJNHEV8G0JXA4MJR6B9FAJ	2019-09-07
17	01HEFJNHFBVT46HRYEFD52Z4C9	2019-09-24
18	01HEFJNHF7YCQ17YMJGK1CQ93R	2019-10-16
19	01HEFJNHF6YXWPF4CB7NRVXVB2	2019-11-10

36. Fetch project_id along with rating.	SELECT t1.project_id, t2.rating FROM project AS t1

	INNER JOIN client_rating AS t2 ON t1.client_id = t2.client_id;

	project_id character varying (100)	rating double precision
1	01HEFJNHERJY9YPAPTBC8NS0CV	4
2	01HEFJNHER60HMXGVTM0DXD1P3	3
3	01HEFJNHER9TPJZF38ZF7VZ7BA	5
4	01HEFJNHER6VA94RGNHYJV5FV9	5
5	01HEFJNHESA8VK58NAKK45ZC6G	5
6	01HEFJNHESSK2TJQSMSY3Z8P5T	2
7	01HEFJNHESCM3BRVZXFTT8MXJM	4
8	01HEFJNHESZ5Y9CF8GBFYZMAQ7	1
9	01HEFJNHES5ANNNRK9MGVX762Q	5
10	01HEFJNHETXD6RKVS3ZCGDR4N4	3
11	01HEFJNHET0W3E3PGP45BHA2TA	2
12	01HEFJNHETZ0DMQD2ET44456KQ	1

37. Fetch details of project by the amount paid by client for the respective project.	SELECT t1.project_id, t2.amount FROM project AS t1 INNER JOIN transactions AS t2 ON t1.client_id = t2.client_id;

Data Output | Messages | Notifications

	project_id character varying (100)	amount character varying (100)
1	01HEFJNHERJY9YPAPTBC8NS0CV	$935.57
2	01HEFJNHER60HMXGVTM0DXD1P3	$963.89
3	01HEFJNHER9TPJZF38ZF7VZ7BA	$847.15
4	01HEFJNHER6VA94RGNHYJV5FV9	$867.37
5	01HEFJNHESA8VK58NAKK45ZC6G	$834.23
6	01HEFJNHESSK2TJQSMSY3Z8P5T	$755.48
7	01HEFJNHESCM3BRVZXFTT8MXJM	$691.82
8	01HEFJNHESZ5Y9CF8GBFYZMAQ7	$789.41
9	01HEFJNHES5ANNNRK9MGVX762Q	$781.36
10	01HEFJNHETXD6RKVS3ZCGDR4N4	$525.23
11	01HEFJNHET0W3E3PGP45BHA2TA	$610.38
12	01HEFJNHETZ0DMQD2ET44456KQ	$548.52
13	01HEFJNHET4TJPYZ6DW0QDGSRT	$824.45
14	01HEFJNHET4B7GWDZYNT595F30	$567.20
15	01HEFJNHETJWX1MC4BRDYB3CZ8	$596.04
16	01HEFJNHEVSA851B4AT2R12K9Z	$565.62
17	01HEFJNHEVX3S8FGBVNCMMSPT9	$605.59
18	01HEFJNHEVBNVHT9M0JQE0VSSK	$725.21
19	01HEFJNHEVDBSPC04E2TJ73NDV	$826.41
20	01HEFJNHEV8G0JXA4MJR6B9FAJ	$636.54
21	01HEFJNHEW966RYTTKMTFP224E	$792.26

Total rows: 99 of 99 | Query complete 00:00:00.064

38. Fetch details of a designer who has worked on the maximum no. of projects in the past.	SELECT *
	FROM designer
	WHERE no_of_projects = (
	  SELECT MAX(no_of_projects)
	  FROM designer
	);

Data Output | Messages | Notifications

	designer_id [PK] character varying (100)	name text	email_id character varying (100)	phone_no bigint	years_of_experience bigint	no_of_projects bigint
1	01HEDA9WT87XRERAM3DQJ6JY5N	Marne Worle…	mworledgek@dailymotion.com	5676905168	15	80
2	01HEDA9WWPJ8T2ZB1RVPYD0YCE	Forest Puttick	fputtick1h@nih.gov	6881048780	17	80

39. Fech project details of a particular designer.	SELECT project_id FROM project WHERE designer_id = '01HEDA9WSW8PFQF6BKKE1YHF1F';

Data Output | Messages | Notifications

	project_id [PK] character varying (100)
1	01HEFJNHETZ0DMQD2ET44456KQ
2	01HEFJNHET4TJPYZ6DW0QDGSRT
3	01HEFJNHET4B7GWDZYNT595F30
4	01HEFJNHETJWX1MC4BRDYB3CZ8
5	01HEFJNHEVSA851B4AT2R12K9Z
6	01HEFJNHEVX3S8FGBVNCMMSPT9

40. Fetch details of all the designers and their past year of experience and no. of projects.	SELECT designer_id, no_of_projects, years_of_experience FROM designer;

Data Output | Messages | Notifications

	designer_id [PK] character varying (100)	no_of_projects bigint	years_of_experience bigint
1	01HEDA9WSDH8A9C51AD91R29DF	67	25
2	01HEDA9WSFP80QC604G0FZ5MGR	31	16
3	01HEDA9WSGY0PHY59P06XREZQ0	36	1
4	01HEDA9WSHH0Z37QMA8BNRP464	2	16
5	01HEDA9WSKQZ9F9WMZ91BNQXR7	77	3
6	01HEDA9WSMFXW4ZN3W56YJE3V7	76	11
7	01HEDA9WSNPWZVG2FV9PCGY5X7	65	18
8	01HEDA9WSQZRBATG8DFFVEM4CB	8	8
9	01HEDA9WSR0P4SCZ99KM72BJPR	76	5
10	01HEDA9WSSQXS20MP15GHTHCJN	58	13
11	01HEDA9WSVP6Y25SXGX5Y4D1YK	49	24
12	01HEDA9WSW8PFQF6BKKE1YHF1F	71	16
13	01HEDA9WSXE90NDVSBDF16QK7N	6	21
14	01HEDA9WSYF0B776C0AA4RFCZV	47	23
15	01HEDA9WT073YBJRTSHNGR4GF2	30	21
16	01HEDA9WT1TXTFY3RG6Z1ZBNMZ	35	20
17	01HEDA9WT2TY1Y48APE4B85SZK	64	17
18	01HEDA9WT4RYZ8H0106SB7TC81	22	10
19	01HEDA9WT53HD0M27ECWHW52...	73	10
20	01HEDA9WT6E94C1Y87QR2W433B	56	7
21	01HEDA9WT87XRERAM3DQJ6JY5N	80	15
22	01HEDA9WT9TZ42AN0F993MP3T2	56	0

Total rows: 100 of 100 | Query complete 00:00:00.059

41. Trigger function to ensure every newly added designer has some experience.	CREATE OR REPLACE FUNCTION check_designer_experience()
	RETURNS TRIGGER AS $$
	BEGIN
	IF NEW.Years_of_Experience <= 0 THEN
	RAISE EXCEPTION 'Designer must have experience greater than 0 years';
	END IF;
	RETURN NEW;
	END;
	$$ LANGUAGE plpgsql;
	
	CREATE TRIGGER check_designer_experience_trigger
	BEFORE INSERT ON DESIGNER
	FOR EACH ROW
	EXECUTE FUNCTION check_designer_experience();

```
CREATE TRIGGER

Query returned successfully in 159 msec.
```

42. Fetch all projects along with their start date and end date.	SELECT P.Project_id, PD.Start_Date, PD.End_Date

	FROM PROJECT P JOIN PROJECT_DETAILS PD ON P.Project_id = PD.Project_id;

Data Output | Messages | Notifications

	project_id character varying (100)	start_date date	end_date date
1	01HEFJNHERJY9YPAPTBC8NS0CV	2021-10-19	2023-08-30
2	01HEFJNHER60HMXGVTM0DXD1P3	2020-04-04	2022-08-08
3	01HEFJNHER9TPJZF38ZF7VZ7BA	2020-10-10	2023-10-14
4	01HEFJNHER6VA94RGNHYJV5FV9	2022-05-28	2023-06-30
5	01HEFJNHESA8VK58NAKK45ZC6G	2023-07-06	2023-07-06
6	01HEFJNHESSK2TJQSMSY3Z8P5T	2020-09-12	2023-12-29
7	01HEFJNHESCM3BRVZXFTT8MXJM	2022-10-18	2022-12-28
8	01HEFJNHESZ5Y9CF8GBFYZMAQ7	2022-09-09	2022-10-02
9	01HEFJNHES5ANNNRK9MGVX762Q	2020-03-17	2023-03-15
10	01HEFJNHETXD6RKVS3ZCGDR4N4	2019-05-01	2022-05-08
11	01HEFJNHET0W3E3PGP45BHA2TA	2019-01-04	2022-04-15
12	01HEFJNHETZ0DMQD2ET44456KQ	2022-01-27	2023-08-13
13	01HEFJNHET4TJPYZ6DW0QDGSRT	2021-05-12	2022-10-25
14	01HEFJNHET4B7GWDZYNT595F30	2021-02-24	2023-05-16
15	01HEFJNHETJWX1MC4BRDYB3CZ8	2021-09-10	2022-10-15
16	01HEFJNHEVSA851B4AT2R12K9Z	2023-01-16	2024-01-07
17	01HEFJNHEVX3S8FGBVNCMMSPT9	2022-07-13	2023-03-07
18	01HEFJNHEVBNVHT9M0JQE0VSSK	2019-01-25	2022-09-03
19	01HEFJNHEVDBSPC04E2TJ73NDV	2021-10-15	2022-09-25
20	01HEFJNHEV8G0JXA4MJR6B9FAJ	2019-09-07	2022-08-28
21	01HEFJNHEW966RYTTKMTFP224E	2023-02-01	2023-06-30
22	01HEFJNHEWKAS7H621JMP2V37F	2022-07-24	2023-03-06

Total rows: 99 of 99 Query complete 00:00:00.049