Questoes do git hub:

1-É um software ou ferramenta que ajuda a rastrear as versões do seu código , o VCS cria copias dos arquivos ao longo do tempo com nomes diferentes e salva automaticamente.

É importante porque facilita muito a vida de um programador que não precisa ter que criar um arquivo zip cada vez que encontra uma versão estável e boa do seu código , o VCS faz esse trabalho por ele e agiliza o processo.

2-O git é um software de versionamento de código , o github é uma plataforma de hospedagem de código que utiliza o git como sistema de controle de versão e utiliza um ambiente web para que desenvolvedores possam colaborar , criar , compartilhar e gerenciar projetos .

Por mais que o Git e o GitHub sejam frequentemente usados em conjunto, da para usar o Git sem o GitHub. Você pode usar o Git em seu ambiente local ou em um servidor interno para controle de versão e colaboração entre desenvolvedores. Mas , o GitHub oferece uma série de recursos e benefícios adicionais, como o armazenamento na nuvem, o compartilhamento fácil de projetos com outros desenvolvedores, a descoberta de projetos de código aberto.

3-No sistema de versionamento distribuído você possui um repositório local e o monitoramento é o tempo todo e não necessita de conexão em tempo real com o servidor como no sistema centralizado.

4- O fluxo de trabalho do Git ocorre da seguinte maneira:

1. Inicialmente, você clona um repositório existente ou inicia um novo repositório em um diretório vazio em seu sistema.
2. Você faz alterações nos arquivos no diretório de trabalho. Essas alterações podem incluir adição, modificação ou exclusão de arquivos.
3. Após fazer as modificações necessárias, você pode usar o comando git add para adicionar os arquivos modificados à área de staging. Isso indica que você deseja incluir essas modificações no próximo commit.
4. Os arquivos adicionados à área de staging agora estão prontos para serem commitados. Você pode usar o comando git commit para criar um novo commit, que registra as modificações realizadas. Esses commits são armazenados no repositório.
5. Após o commit, os arquivos no repositório refletem o estado do projeto naquele momento. As alterações no diretório de trabalho agora podem ser modificadas novamente, criando um novo ciclo de modificação, staging e commit.

5-A tabela clientes possuí dados multivalorados , telefone 1 2 e 3 , isso não é uma pratica boa e esses dados deveriam estar em uma outra tabela , pois a pessoa pode ter mais números de telefone , é melhor padronizar com apenas um ou criar essa outra tabela .Outro erro que se pode perceber é o uso de atributos compostos na coluna de endereço\_fisico , porque fica muito vago o que a pessoa pode inserir como dado dentro da coluna , e seria melhor separar em colunas mais específicos como em : bairro , rua , numero e etc

6-

A tabela clientes faz um relacionamento N:N com a tabela lojas por meio da tabela pedidos

A tabela clientes faz outro relacionamento N:N com a tabela lojas por meio da tabela envios

A tabela lojas faz um relacionamento N:N com a tabela produtos por meio da tabela estoques

A tabela pedidos faz um relacionamento N:N com a tabela produtos por meio da tabela pedidos itens

É um relacionamento N:N porque existem múltiplos registros nas tabelas que se associam por meio de uma tabela intermediaria (definição geral)

7-Não esta incorreto, porque uma PK e PFK em uma mesma tabela .

8-Porque as FKs da tabela pedidos e produtos fazem parte da chave primaria da tabela pedidos\_itens e por isso o relacimento é identificado

Já com a tabela envios a FK da tabela envios não faz parte da chave primaria da tabela pedidos\_itens e por isso o relacionamento é não identificado

9- Relacionamento N:N , Porque pra existir esse relacionamento tem que existir outra tabela alternativa que guarda outros tipos de dados que se relacionam com as duas tabelas.

Existem 4 relacionamentos N:N , não identifiquei nenhum erro nos relacionamentos .

10-É um tipo de dado que pode armazenar dados binários como imagens , multimidia e arquivos pdf .

Esse tipo de dados é utilizado para armazenar dados não estruturados

11-um banco de dados é uma coleção organizada de informações estruturadas ou não .

O usuário é quem administra/interage com o banco de dados, responsável por inserir , excluir , atualizar ,modificar dados .

Schema é uma estrutura logica que ajuda a organizar e agrupar os objetos(tabelas,relações,procedimentos) do banco de dados

12-É muito importante para você organizar o sistema logico do seu banco de dados : Evita conflito de dados nas tabelas, facilita a gestão e manutenção , principalmente em sistemas mais complexos ,controle de acesso e segurança : você pode definir permissões com base nos schemas criados.

Em empresas medias e grandes é obrigatório o uso de schemas por existirem vários departamentos e varias semelhanças entre os tipos de dados e etc

13-Os objetos do banco de dados vão para um schema publico , é ruim, porque um schema publico não identifica o objetivo daqueles dados armazenados ali , além de poder haver ambiguidades com objetos criados em outro schemas ,e em bancos de dados maiores o gerenciamento e a manutenção do banco de dados fica mais difícil.

14-Gostei muito da proposta do projeto , me dediquei do inicio ao fim e fiquei muito empenhado enquanto fazia(uma coisa bem incomum na verdade ), aprendi muito sobre Git e Git hub e me aprofundei muito na linguagem SQL.

15-Gostei muito do Postgres , embora não mude muito em relação ao MariaDB , de qualquer forma por ser o SGBD que eu mais me aprofundei(Postgres) , gostei muito do modelo de comandos , a dinamicidade dos comandos , o aprofundamento em relação aos objetos do banco dados me fez entender realmente como funciona um SGBD e consegui tirar muitas de minhas duvidas.

Já o Oracle eu ainda não me aprofundei muito na linguagem administrativa e nem nos comandos de criação.