**Teste - Analista de Desenvolvimento Júnior - Suporte**

**Questão 1:** Considere a seguinte estrutura de tabelas:

| **Pacientes** | **Atendimento** | **Prontuário** |
| --- | --- | --- |
| id\_paciente | id\_atendimento | id\_prontuario |
| nome\_paciente | id\_paciente | id\_paciente |
| tel\_paciente | data\_atendimento | id\_atendimento |
| email\_paciente | hr\_atendimento | descricao\_prontuario |
| status\_paciente | nome\_unidade |  |
| data\_nascimento | endereco\_unidade |  |
|  | cd\_especialidade |  |

1. Escreva uma consulta SQL que retorne todos os atendimentos do paciente com a especialidade "Neurologia" nos últimos 6 meses.

**SELECT \***

**FROM Pacientes p**

**JOIN Atendimento a ON p.id\_paciente = a.id\_paciente**

**WHERE a.cd\_especialidade = 'Neurologia'**

**AND a.data\_atendimento >= DATE\_SUB(CURDATE(), INTERVAL 6 MONTH)**

**AND a.data\_atendimento <= CURDATE();**

1. Suponha que você precise obter todos os prontuários associados ao paciente de ID 123, incluindo os detalhes do atendimento relacionado a cada prontuário. A consulta deve retornar o nome do paciente, a descrição do prontuário e a data do atendimento correspondente.

**SELECT p.nome\_paciente, pr.descricao\_prontuario, a.data\_atendimento**

**FROM Pacientes p**

**JOIN Prontuario pr ON p.id\_paciente = pr.id\_paciente**

**JOIN Atendimento a ON pr.id\_atendimento = a.id\_atendimento**

**WHERE p.id\_paciente = 123;**

**Questão 2:** Um cliente relata que está enfrentando lentidão em um determinado recurso do sistema. Como você abordaria esse problema para identificar a causa raiz e propor uma solução?

**Primeiramente pediria informações detalhadas sobre o recurso que estava apresentando a lentidão e em que parte estava ocorrendo (ao abrir, ao executar determinada tarefa do recurso e etc). Após isso e sabendo o processo que estava sendo feito pelo cliente, tentaria replicar o exato processo em um ambiente controlado para verificar se a lentidão também ocorreria nesse ambiente e, caso ocorresse, ficaria mais fácil determinar onde está o problema. Caso a lentidão não ocorresse no nosso ambiente, provavelmente significa que era algum erro de versão ou específico da máquina do cliente, o que já eliminaria muitas possibilidades. Em última instância, pediria acesso remoto ao cliente para verificar de forma clara onde estava o erro e porquê estava ocorrendo. Após verificar isso, entraria com a solução com base no erro, mas que provavelmente seria algo que seria resolvido com um script pontual.**

**Questão 3:** Um usuário informa que está recebendo uma mensagem de erro ao tentar realizar uma operação específica no sistema. Explique como você investigaria esse problema e quais etapas seguiria para solucioná-lo.

**Em um primeiro momento pediria informações detalhadas do erro enfrentado pelo cliente (prints, a própria mensagem de erro, processo até a mensagem). Verificaria também os logs do sistema que provavelmente dariam um “norte” para ver em que momento o erro ocorreu e se haviam erros relacionados. Além disso, como na questão anterior, tentaria replicar o erro a partir do mesmo processo que ele fez e, caso o erro fosse visto também no teste, verificaria o código do recurso relacionado ao erro e corrigiria o mesmo. Caso o erro não fosse mostrado mesmo após testar em vários tipos de ambientes, a constatação óbvia será de a causa do erro está na máquina do cliente, seja na versão do sistema instalado ou problemas de configuração e etc. Para resolver, caso fosse esse o caso, pediria permissão ao cliente para o acesso remoto e então resolveria o problema a partir daí, reinstalando o sistema e fazendo um debug até encontrar o erro.**

**Questão 4:** Descreva como você lidaria com uma situação em que um usuário encontra um erro desconhecido.

**Caso o usuário encontrasse um erro que até então nunca aconteceu, pediria a ele as informações detalhadas como sempre (mensagem de erro, prints relacionados e o processo detalhado até o erro). Nesse caso em específico, provavelmente o erro existiria também no nosso sistema, o que tornaria a resolução mais fácil, uma vez que saberíamos que o erro é no próprio sistema e a partir daí a correção seria feita no código do mesmo no recurso em que estivesse o erro. Porém, caso após vários testes fosse constatado que o erro se encontra aparentemente apenas na máquina do cliente, realizaria o mesmo processo de acesso remoto dito nas outras questões.**

**Questões de Lógica:** Usando uma das seguintes linguagens de programação: C, C++, Java, C#, Javascript ou Python, resolva as questões a seguir.

**Questão 1:** Faça um programa, em seguida, um programa que solicite a entrada de dois números, em seguida imprime na tela o quadrado do menor número e a raiz quadrada do maior número, se for possível.

#include <stdio.h>

#include <math.h>

int main() {

double numero1, numero2;

printf("Digite o primeiro número: ");

scanf("%lf", &numero1);

printf("Digite o segundo número: ");

scanf("%lf", &numero2);

double menorNumero, maiorNumero;

if (numero1 < numero2) {

menorNumero = numero1;

maiorNumero = numero2;

} else {

menorNumero = numero2;

maiorNumero = numero1;

}

int quadradoMenor = (int)(menorNumero \* menorNumero);

int raizQuadradaMaior = maiorNumero >= 0 ? (int)sqrt(maiorNumero) : -1;

printf("A raiz quadrada do menor número é: %d\n", quadradoMenor);

if (maiorNumero >= 0) {

printf("A raiz quadrada do maior número é: %d\n", raizQuadradaMaior);

} else {

printf("forneça um número válido.\n");

}

return 0;

}

**Questão 2:** Faça um programa que solicite ao usuário o ano em que ele nasceu e verifique se o ano é Bissexto ou não e imprima uma mensagem na tela.

#include <stdio.h>

int main() {

int ano;

printf("Digite o ano em que voce nasceu: ");

scanf("%d", &ano);

if ((ano % 4 == 0 && ano % 100 != 0) || ano % 400 == 0) {

printf("%d voce nasceu em um ano bissexto.\n", ano);

} else {

printf("%d voce nao nasceu em um ano bissexto.\n", ano);

}

return 0;

}