

Q1	
Q2	
Q3a.	
Q3b.	
Q3c.	
Q3d.	

Fundação CECIERJ – Vice Presidência de Educação Superior à Distância
Curso de Tecnologia em Sistemas de Computação
Disciplina: Programação II
AP2 –1º Semestre de 2014

Nome: _____

Questão 1 (2 ptos.)

Descreva a funcionalidade do código a seguir. Sua resposta deve descrever em profundidade o que o código está fazendo, para que serve tal código, e em especial os comandos sublinhados.

```
<?php
    if ( isset( $_COOKIE['visitaRecente']) ){
        $msg = "Ficamos felizes com seu retorno";
    }
    else{
        $msg = "Seja bem vindo";
    }
    setcookie("visitaRecente", true, time()+60*60*24*30);
?>

<html>
<head> <title> Nosso site </title> </head>
<body>
<h1> <?php echo $msg; ?> </h1>
</body>
</html>
```

Resposta:

O código acima identifica se o usuário está retornando à página ou se é a primeira vez que a acessa nos últimos 30 dias. Para fazer isso, o código faz uso de duas funções nativas do PHP. A primeira delas é a função setcookie, que cria ou atualiza um cookie com nome “visitaRecente” e valor “true” com validade de 30 dias a partir da data corrente do servidor.

O bloco de código logo acima da função setcookie é uma condição baseada no retorno da função isset(\$_COOKIE['visitaRecente']). Essa função verifica se o cookie de nome “visitaRecente” está criado no computador do usuário e está dentro da data de validade. Em outras palavras, se tiver menos de 30 dias que o usuário acessou a página pela primeira vez, ele verá a mensagem “Ficamos felizes com seu retorno” impressa na tela. Se tiver mais de 30 dias desde seu último acesso, ou se ele nunca tiver acessado a página antes, ele verá a mensagem “Seja bem vindo”.

Questão 2 (2 ptos.)

A existência de um cadeado no navegador (na barra de tarefas ou ao lado do endereço da página) demonstra a certificação de página segura (SSL - Secure Sockets Layer). Descreva o que é essa segurança, para que serve e dê exemplos de sua utilização.

Resposta:

O SSL é um protocolo de segurança de redes que permite uma troca segura de mensagens entre cliente e servidor de forma segura. A segurança é obtida através de uma criptografia que é aplicada às mensagens que são enviadas pelo servidor para o usuário e vice versa.

Ela funciona da seguinte maneira: quando o cliente solicita a conexão com um site seguro, o site envia seu Certificado Digital e verifica se seu o certificado é confiável. Uma vez que as informações tenham sido confirmadas, uma chave pública é enviada pelo servidor e as mensagens podem ser trocadas. Uma mensagem que tenha sido criptografada com uma chave pública somente poderá ser decifrada com a sua chave privada (simétrica) correspondente.

Ou seja, o servidor envia uma chave pública, que deve ser usada pelo cliente para criptografar as mensagens que irá enviar para o site. Somente o site possui sua chave privada, e deverá usá-la para descriptografar a mensagem enviada pelo usuário. A mensagem final só fara sentido de a chave privada do servidor for legítima, garantindo, assim, que a mensagem só poderá ser lida pelo destinatário correto.

Um exemplo de uso do protocolo SSL é nos sites de banco, ou sites de compras online.

Questão 3 (6 ptos.)

Descreva um script PHP denominado "**questão3.php**" para permitir a entrada de dados via um formulário (questão 3a.), a validação dos dados digitados (questão 3b.) e a inserção dos dados em um banco de dados (questão 3c.). Cada uma dessas três tarefas deve ser separada em uma função e chamada segundo a lógica descrita na questão 3d., conforme detalhado a seguir.

a. (1.0 pto)

Desenvolva uma função chamada **exibeFormulário** que produz um formulário HTML permitindo a entrada de um nome, um email e uma data de nascimento. O processamento do formulário deve ser associado ao script "**questão3.php**".

b. (2.0 ptos)

Desenvolva uma função chamada **validaEntradas** responsável por garantir as seguintes regras:

- nenhum dos três campos digitados no formulário pode estar vazio;
- o nome digitado deve possuir pelo menos 3 caracteres;
- a data deve estar no formato **dd/mm/aaaa**;

- o email digitado deve conter o símbolo de @ e não deve conter espaços.
Sua função deve retornar verdadeiro caso as regras sejam atendidas, ou falso caso contrário. No caso de erro, deve também exibir em HTML uma mensagem informando qual o erro ocorrido.

c. (2.0 ptos)

Desenvolva uma função chamada **armazenaDadosNoBanco**. Esta função é responsável por:

- Abrir conexão com o servidor **CEDERJ** com o usuário **prog2** e senha **prog2014_1** e testar se foi aberta com sucesso;
- Selecionar o banco de dados **CURSO_TSC** e testar se foi selecionado com sucesso;
- Verificar na tabela **Cadastro** a existência prévia de um usuário com mesmo **nome** e **data de nascimento**; Se existir, deve exibir a mensagem: "Usuário já cadastrado" e não continuar o registro;
- Inserir na tabela **Cadastro** uma nova linha, contendo o **nome**, **data de nascimento** e **email**;
- Fechar a conexão com o servidor **CEDERJ**.

Sua função deve retornar verdadeiro caso todas as tarefas apresentadas sejam realizadas com sucesso, ou falso caso contrário.

d. (1.0 ptos)

Escreva a lógica responsável por decidir entre mostrar o formulário ou validar e armazenar os dados.

A decisão deve observar a existência não nula da variável **nome**. Caso a variável **não** exista ou possua valor nulo, considere o estágio de entrada de dados via formulário, chamando a função **exibeFormulário**.

Caso contrário, o script deve passar a processar a validação dos dados chamando a função **validaEntradas**.

Se a validação não for realizada com sucesso, o script deve voltar a exibir o formulário chamando a função **exibeFormulário**.

Uma vez aprovada a validação, o script passa a executar a inserção no banco de dados chamando a função **armazenaDadosNoBanco**. Por fim, observando o retorno da função **armazenaDadosNoBanco** deve exibir em HTML a mensagem: "Usuário cadastrado com sucesso" ou "Erro no cadastrado do Usuário".

Resposta:
questão3.php:

```
<!-- Código da letra A -->
<html>
<head>
    <title>T&iacute;tulo do Formul&aacute;rio</title>
    <?php
function exibeFormulario() {
    echo "<form action='./questao3.php' method='post'>";
    echo "    <label for='nome'>Nome:</label>";
    echo "    <input type='text' name='nome' /><br />";
    echo "    <label for='email'>Email:</label>";
    echo "    <input type='text' name='email' /><br />";
    echo "    <label for='nascimento'>Data de Nascimento:</label>";
    echo "    <input type='text' name='nascimento' /><br />";
    echo "</form>";
}

//Aqui entra o código da letra B

//Aqui entra o código da letra C
?>
</head>
<body>
    <?php
        //Código da letra D
        if (isset($_POST['nome'])) {
            if (!(validaEntradas() and armazenaDadosNoBanco())) {
                unset($_POST['nome']);
            }
        }

        if (!isset($_POST['nome'])) {
            ?><h1>Formul&aacute;rio</h1><?php
            exibeFormulario();
        }
    ?>
</body>
</html>
```

```

//Código da letra B
function validaEntradas() {
    $valida = true;

    //Nenhum campo pode estar vazio
    if (($_POST['nome'] == '' ||
        $_POST['email'] == '' ||
        $_POST['nascimento'] == '' ) ||
        //O nome deve ter pelo menos 3 caracteres
        (strlen($_POST['nome']) < 3) ||
        //A data deve estar no formato dd/mm/aaaa
        (!preg_match('/^[0-9]{2}/[0-9]{2}/[0-9]{4}$/',
            $_POST['nascimento']))) ||
        //O email digitado deve conter o símbolo de @
        //e não deve conter espaços
        (strpos($_POST['email'], '@') !== true ||
            strpos($_POST['email'], ' ') !== false)) {

        echo "<p>Falha ao validar formulário</p>";

        return false;
    }
    return true;
}

```

```

//Código da letra C
function armazenaDadosNoBanco($nome, $email) {
    //Abrir conexão com o servidor CEDERJ com o usuário prog2
    //e senha prog2014_1 e testar se foi aberta com sucesso
    if (!mysql_connect('CEDERJ', 'prog2', 'prog2014_1')) {
        return false;
    }

    //Selecionar o banco de dados CURSO_TSC e testar se foi
    //selecionado com sucesso
    if (!mysql_select_db('CURSO_TSC')) {
        mysql_close(); return false;
    }

    //Verificar na tabela Cadastro a existência prévia de um
    //usuário com mesmo nome e data de nascimento; Se existir,
    // deve exibir a mensagem: "Usuário já cadastrado" e não
    //continuar o registro
    $query =
        "SELECT 1 FROM Cadastro WHERE nome = '$nome' OR " +
        "email = '$email' LIMIT 1;";

    $resultado = mysql_query($query);

    if (mysql_num_rows($resultado) > 0) {
        echo "Usuário já cadastrado.";
        mysql_close();
        return false;
    }

    //Inserir na tabela Cadastro uma nova linha, contendo o
    //nome, data de nascimento e email;
    $query =
        "INSERT INTO Cadastro VALUES (DEFAULT, '$nome', " +
        "'$email', '$nascimento');";

    if (!mysql_query($query)) {
        echo "Erro ao cadastrar usuário.";
        mysql_close();
        return false;
    }

    mysql_close();

    return true;
}

```