Alguns dizem que atualmente os iframes não tem mais utilidade e que caíram em desuso, entretanto isso está errado e os iframes não estão defasados, sendo que podem sim serem úteis.

É importante que você não confunda os "iframes" com os "frames"... Não usamos mais os frames em HTML5, é muito difícil, só em casos muito pontuais. Os frames estão realmente bem defasados, mas os iframes não.

Observação: Esse " i " na palavra iframe é derivado da palavra " inline " (em linha). Dessa forma, resumidamente um iframe seria um "frame inline", ou também podemos chamar de "quadro em linha" (frame = quadro/moldura + inline = em linha).

Para que serve um iframe?

Basicamente um iframe serve para criarmos uma janela, como se fosse uma aba nova de navegador, só que dentro da sua página. Uma nova instância de navegador (nova aba) dentro do seu site.

Dessa forma, o iframe é uma tag em um documento HTML que viabiliza a inclusão de outro documento HTML dentro do primeiro. Em outras palavras, ele permite que, a partir de uma página inicial, seja possível abrir uma seção de outra página qualquer (aba).

Usando iframes:

Para criar um iframe em HTML, usamos a tag " <iframe></iframe> ". Após abrir a tag, uma estrutura em forma de quadro será criada, essa estrutura é do tipo "inline"... Isso significa que caso você insira a tag no meio de uma linha/frase por exemplo, esse quadro ficará posicionado exatamente entre as palavras da frase. Não terá um comportamento de caixa "box-level" a menos que você altere o display do elemento.

De forma padrão, ao abrir a tag "iframe", os parâmetros/atributos " src " e " frameborder " já estarão embutidos na tag.

O atributo frameborder vem com o valor "0", removendo a borda do nosso iframe.

Já como valor do atributo src, você deve inserir o link da página que deseja abrir dentro da janela do iframe.

Dessa maneira o site escolhido será carregado dentro desse iframe. De forma padrão o iframe é carregado em uma dimensão de altura e largura muito pequena, entretanto podemos modificar isso posteriormente.

Nota: Existem alguns sites que possuem proteção para que o seu conteúdo não seja exibido dentro de um iframe em páginas de terceiros, nesse caso a conexão será recusada. Um exemplo de site que não aceita conexão com iframes é o Google.

Mais adiante iremos nos aprofundar na temática da proteção nos iframes. Iframes são úteis em determinadas situações, mas você precisa saber que eles tem alguns problemas de segurança... Entretanto, com o avanço do HTML5, é possível prever e resolver (ou pelo menos reduzir) tais inconvenientes na questão da segurança.

Por isso, é importante que você saiba usar o iframe, e que também saiba em que momentos deve usar ou não usar. Quando utilizar, saber quais são as facilidades que ele vai te trazer e quais são os problemas que ele pode te trazer.

TRATAMENTO DE ERRO EM IFRAMES:

Nem todos os navegadores para celular, para dispositivos móveis, tem suporte para iframes...

No meio da tag (<iframe><p>aqui</p></iframe>) podemos inserir conteúdo - texto - no entanto, esse texto somente ficará visível caso ocorra um "erro", caso o navegador do dispositivo não tenha compatibilidade com iframes (igual nas tags de video)

Portanto, dentro da tag iframe, podemos deixar uma mensagem do tipo: " Seu navegador não é compatível com essa funcionalidade " - Que atuará como uma espécie de tratamento de erro que será exibido somente em caso de falha no carregamento do iframe para os usuários com navegadores incompatíveis com esse elemento (alguns dispositivos móveis).

Também, ao invés de colocar um aviso no meio da tag, podemos inserir um link que direciona o usuário para o site que ele deveria ter visualizado inicialmente pelo iframe (insira o nome do respectivo site na etiqueta do link - texto do link).

Dessa forma, se por acaso o seu iframe não for compatível com o navegador do usuário, o seu tratamento de erro (mensagem ou link) será exibido para ele.

ESTILIZAÇÃO DE IFRAMES:

Referente às propriedades de largura e altura, por padrão essas dimensões em iframes são respectivamente de 300x150px.

Se você quiser, pode configurar essas medidas de largura e altura diretamente na tag do iframe. Para fazer isso, existem os parâmetros width e height, que nesse caso devem ser inseridos dentro da tag de abertura do iframe. (não é necessário abrir style)

Você também pode configurar largura e altura usando CSS, para fazer isso basta abrir um seletor para o elemento iframe e inserir as declarações.

Observação: Nesse caso das CSS, as configurações feitas no seletor tem preferência e sempre irão sobrepor as configurações de largura e altura feitas por parâmetros HTML, lembre-se disso. Caso não exista nenhuma configuração nas suas folhas de estilo, aí sim ficam valendo as dimensões que foram configuradas no HTML.

Para o elemento iframe, também existe o parâmetro " scrolling ", que igualmente deve ser inserido dentro da tag de abertura do iframe. Esse parâmetro serve para configurar a barra de rolagem do iframe, e ele aceita os seguintes valores: "auto", "yes" e "no".

auto - É o valor padrão, faz com que a barra de rolagem no iframe apareça e possa ser utilizada normalmente.

yes - Força a exibição de uma barra de rolagem, mesmo que o site embutido no iframe não tenha uma.

no - Bloqueia a barra de rolagem do site que está embutido no iframe, impedindo o usuário de scrollar.

Você pode também adicionar uma borda com o estilo que preferir no seu iframe, para fazer isso basta inserir a declaração de borda dentro do seletor CSS do iframe

Nota: Como nós já sabemos, vários sites possuem proteção contra o iframe, bloqueiam a sua visualização dentro deles... Talvez você possa pensar que por causa disso os iframes não tem serventia e acabe se perguntando de que forma vai utilizá-los... A resposta é simples: Você também pode inserir o seu próprio conteúdo (uma página sua, local) dentro de um iframe, existem muitas possibilidades com isso.

Inserindo conteúdo de site local no iframe:

1) Criar na mesma pasta, uma página com o conteúdo a ser exibido no iframe da página principal;

2) No iframe da página principal, inserir em 'src' o link da página a ser exibida;

Ex: <iframe src="pag001 . html" frameborder="0"></iframe>

Obs: Um site com tabela também pode ser inserido dentro de um <iframe>, mas é recomendável inseri-la dentro de uma

<div>, pois facilita a ação dos mecanismos de busca;

NAVEGAÇÃO DE ARQUIVOS:

Quando exibimos conteúdo de forma local pelo nosso iframe, às vezes é interessante criar uma pasta para armazenar justamente essas páginas extras que só existem para serem exibidas pelo iframe.

Por exemplo: Dentro do nosso servidor, podemos criar uma pasta chamada "paginas-extras" e armazenar todas as páginas que serão exibidas pelo iframe nessa pasta (lembre-se que ao armazenar as páginas em outra pasta, o endereço local no iframe deve ser atualizado adicionando o caminho da pasta).

IFRAME INTERATIVO:

Quando você cria um iframe para exibir uma página de seu interesse, eventualmente você pode querer substituir essa página que está configurada no iframe por outra... Naturalmente, para alterar a página que está sendo exibida, você muda o endereço contido no parâmetro src do iframe.

Entretanto, também é possível realizar essa transição de página de forma interativa, atribuindo essa ação para o próprio usuário.

Por exemplo, considerando que você tenha uma página primária com um iframe dentro, e que também tenha mais três outras páginas secundárias:

Para fazer a transição interativa dessas três páginas no seu iframe, é necessário que você, lá na página primária, monte uma lista com os links das três páginas secundárias. Exemplo:

- Primeira página

- Segunda página

- Terceira página

A partir daí, queremos que quando o usuário clique em algum desses links, a respectiva página secundária seja renderizada no iframe (e não será necessário recarregar a página primária para que isso ocorra).

Para tal, na tag do iframe você precisa inserir o parâmetro " name ", e atribuir a ele uma nomenclatura (de sua preferência - ex: 'frame')

Agora, para associarmos os nossos links de páginas secundárias ao iframe, devemos inserir na tag dos links (<a></a>) o parâmetro "target", tendo como valor justamente a nomenclatura do parâmetro "name" ('frame' no caso do nosso exemplo).

Feito isso, ao clicarmos em algum link da lista, a respectiva página secundária será carregada no iframe. Dessa forma conseguimos criar um sistema de navegação numa página primária e interagir abrindo páginas secundárias dentro do iframe... E isso tudo apenas usando links com o target apontando para o "name" do nosso iframe (lembre-se que para fazer isso o parâmetro name é obrigatório - e esse name tem que ser exatamente o mesmo que usamos no target)

Nota: Inicialmente, quando a nossa página primária for carregada, o nosso iframe estará vazio... Isso se deve ao fato de que não especificamos nenhuma página dentro do parâmetro src dele. Se você quiser que inicialmente (antes de clicarmos em qualquer link da lista de páginas secundárias) o iframe seja carregado com alguma página, é só inserir o endereço no parâmetro src do iframe.

PARÂMETRO NAME:

Sobre o parâmetro "name", de certa maneira ele serve para identificar um elemento, assim como o parâmetro "id" faz...

De forma geral, pode-se dizer que o id é importante para o browser (front-end), para a manipulação do DOM (arvore do documento) e CSS; e o name é importante para o servidor (back-end).

O name é principalmente usado para elementos de <form>, ou seja para campos de inserção de dados e é exatamente o name que identifica no servidor o campo. Quando o envio de um form é executado, os campos dele vão para o servidor num objeto cujas chaves são o name que esses elementos têm.

E diferente do "id", pode haver mais do que um elemento com o mesmo "name"...

Uma outra funcionalidade do parâmetro "name" é a seguinte: Ao invés de atualizar a página, fazer uma submissão de formulário para um iframe... Ainda que o action esteja apontando para outra página, a renderização da mesma se dará dentro do iframe. (usamos o name para isso nesse exercício)

Mas como foi dito, geralmente o atributo name serve para representar uma coleção de valores, enviados através de um formulário, para o servidor... Algo um pouco mais avançado.

Inicialmente, quando a nossa página primária for carregada, o nosso iframe interativo estará vazio. Isso se deve ao fato de que não especificamos nenhuma página dentro do parâmetro src dele...

O ideal seria que colocássemos algum conteúdo dentro dele para ser exibido de forma inicial, antes que alguma página secundária seja renderizada de forma interativa.

Uma das maneiras de se fazer isso é você criar um arquivo separado e inserir o endereço desse arquivo no parâmetro src do iframe, funcionando como uma página padrão e inicial.

Essa página inicial pode conter informações como: " Escolha um das opções acima (páginas secundárias) " e alguma imagem em forma de logo.

Entretanto, para um iframe interativo, também é possível fazer essa "página inicial" com o parâmetro " srcdoc ". Esse parâmetro permite que adicionemos, como valor do parâmetro, especificações em HTML (h1, p, img e por aí vai).

Esse código HTML que inserirmos como valor de srcdoc será renderizado no nosso iframe interativo, funcionando justamente como uma página inicial.

Quando você decide usar o parâmetro srcdoc, a primeira coisa a se fazer é deletar o parâmetro padrão "src", e aí sim inserir o srcdoc... Substituir o src pelo srcdoc. Posteriormente insira o seu código HTML como valor de srcdoc.

Exemplo:

<iframe srcdoc="<h1>Escolha uma das opções acima</h1> <p>Lorem ipsum dolor sit amet consectetur adipisicing elit. Nobis sapiente et atque aperiam</p> <img src='imagens\cursoemvideo-logo-branca.png'>" name="frame"><p>Infelizmente o seu navegador não é compatível com essa funcionalidade</p></iframe>

Dessa maneira geramos uma página inicial de forma estática para o nosso iframe sem precisar criar um arquivo separado.

Entretanto, o srcdoc tem a desvantagem que só podemos inserir um código básico em HTML, não dá pra fazer muita coisa além disso. Por isso a recomendação é que, caso você queira algo mais evoluído, fazer como foi dito anteriormente e criar um arquivo separado com o seu código HTML e inclusive CSS. Ao invés de criar um arquivo separado, você também pode inserir uma imagem como plano de fundo, background... É uma opção.

Inconvenientes do iframe

- Dificuldades de acesso ao conteúdo pelos mecanismos de busca;

- Baixa acessibilidade (dificuldades de acesso ao conteúdo pelos leitores de tela, para deficientes visuais);

- Baixa usabilidade (navegador ou usuário acabam se confundindo);

- Baixa segurança (abre as portas do seu site para outros sites);

Existem algumas configurações que podemos fazer nos nossos iframes para torná-los um pouco mais seguros em relação a algumas "falhas" e inconvenientes que eles possuem. Dessa forma, podemos proteger um pouco mais o nosso site no que diz respeito aos ataques de "cross-site scripting" e de "xss" mencionados anteriormente.

Como já foi dito, ao inserir sites de terceiros num iframe, pode ocorrer dessas páginas externas usarem o seu site para aplicarem técnicas de captura de informações do usuário... Uma das medidas de proteção que você pode fazer é justamente bloquear isso.

PARÂMETROS DE SEGURANÇA EM IFRAMES:

SANDBOX:

Para executar esse bloqueio, é necessário inserir alguns parâmetros de segurança na tag do seu iframe. Tenha em mente que os ataques citados aqui, inicialmente dependem que o usuário envie seus dados a partir de um formulário... Dessa maneira, para os sites de terceiros inseridos no nosso iframe, devemos bloquear justamente essa função de envio de dados por formulários, neutralizando todas as ações que são potencialmente maliciosas nesses sites.

Para fazer isso, na tag do iframe, usamos o parâmetro " sandbox "... O valor para esse parâmetro deve ser o " sandbox " (sim, o mesmo nome: sandbox recebe sandbox).

Ao inserirmos esse parâmetro com esse valor, ele bloqueia todo e qualquer acesso que outros sites (que estejam dentro do iframe) tentem fazer... Dessa maneira, no nosso iframe, os formulários desses sites não irão executar a ação de envio de dados... Mesmo que o usuário ainda consiga preencher os campos de formulário, ao clicar em "enviar", o comando simplesmente não é executado, não é aceito.

Isso porque quando usamos essa configuração ( sandbox="sandbox" ) criamos uma "proteção" um pouco maior em relação a ataques, não estaremos mais capturando nenhum tipo de informação.

Quando adicionamos o sandbox com o valor sandbox em nosso iframe, automaticamente o site que está dentro do quadro perde algumas funcionalidades, dentre elas:

- Não pode mais enviar dados de formulários

- Não pode mais executar scripts

- Desabilita todo tipo de API, janelas modais e popups

- Desabilita todo tipo de plugin com <embed>, <object>, <applet>

- Evita que o site dentro do iframe assuma a navegação top level do navegador

- Bloqueia recursos como autoplay e foco automático em elementos de formulário

REFERRERPOLICY:

Uma outra maneira de deixar o seu site um pouco mais seguro, é utilizando também o parâmetro " referrerpolicy ", que serve para configurar a política de referência. O valor padrão usado para esse parâmetro é o " no-referrer-when-downgrade ".

Agora, se você quiser tornar aquilo que está carregado dentro do iframe um pouco mais limitado, ao invés de usar o valor padrão, é só usar o valor " no-referrer ".

Sem entrar muito em termos técnicos, normalmente os sites conseguem monitorar nossos rastros de navegação através de um campo no cabeçalho http chamado Referer Header. Ele sempre indica onde o visitante estava quando decidiu ir para o seu site.

Esse é um dos recursos que as redes sociais como Twitter, Facebook e Instagram e sites como Google e YouTube usam para capturar os seus dados de navegação e entender o seu padrão de navegação para mostrar propagandas. Mas não é só pra isso que serve...

Acontece que, quando usamos essa configuração ( referrerpolicy="no-referrer" ), informamos que aquilo que está carregado dentro do iframe, NÃO vai coletar nenhum tipo de informação do usuário. O conteúdo que está dentro do iframe não vai poder capturar dados de ninguém, tonando o ambiente o mais anônimo possível. Conseguimos definir quantos dados serão compartilhados com o site de destino... Ou seja, com o valor "no-referrer", nenhum dado.

Existem outros valores para o parâmetro referrerpolicy, como:

- no-referer-when-downgrade (valor padrão citado)

- origin

- origin-when-cross-origin

- same-origin

- strict-origin

- strict-origin-when-cross-origin

- unsafe-url

Dessa forma, esses dois parâmetros de segurança (sandbox e referrerpolicy), com os valores orientados, irão ligar o máximo de segurança dentro de seu iframe... Isso é bastante recomendado.

VALORES DE EXCEÇÃO PARA OS PARÂMETROS DE SEGURANÇA:

Porém, vamos imaginar que em um caso muito específico, para o funcionamento correto da sua aplicação, você queira deixar com que os iframes executem o envio de dados através de formulários... Nesse caso, devemos abrir algumas exceções no parâmetro "sandbox", fazemos isso com valores que concedem certos tipos de autorizações específicas.

Como sabemos, o parâmetro sandbox com o valor sandbox liga todas as configurações de segurança no máximo, aciona o máximo nível de segurança que ele consegue fazer.

Entretanto, conseguimos abrir exceções e desligar algumas coisas. Segue alguns valores para o parâmetro " sandbox " que concedem autorizações de segurança específicas (exceções de segurança):

allow-same-origin: Para que formulários que estão em páginas do nosso próprio servidor (conteúdo local) passem a funcionar no iframe, podemos inserir o valor " allow-same-origin "... Dessa forma, uma exceção nas configurações de segurança é feita para os conteúdos de mesma origem (páginas do nosso próprio servidor passam a funcionar normalmente com permissão total, enquanto páginas de terceiros são "neutralizadas").

allow-forms: Também podemos conceder uma exceção para permitir que todo e qualquer formulário passe a funcionar normalmente (não só de conteúdo local, mas de páginas de terceiros também). Para permitir isso, use o valor " allow-forms ".

allow-scripts: O valor " allow-scripts " serve para abrirmos uma exceção na segurança para permitirmos scripts - Quando usamos especificamente o valor citado acima (allow-forms) temos também que usar justamente o "allow-scripts" em conjunto, visto que o HTML5 executa scripts quando roda formulários. Formulários necessitam de scripts para operar. Dependendo do site externo, é arriscado deixar scripts serem executados e ativos, isso se deve porque scripts podem ser maliciosos, então cuidado.

Ob.: Referente aos valores de exceção para sandbox descritos acima, uma coisa que podemos fazer é colocar dois valores ou mais para o mesmo parâmetro sandbox... Basta separar os valores com um espaço e manter dentro das aspas, dessa forma várias exceções de segurança serão aplicadas ao mesmo tempo. EX: sandbox="allow-same-origin allow-forms allow-scripts"

Atenção: Apesar de podermos abrir essas exceções de segurança, tudo aquilo que não foi especificado como exceção, continua ligado... Ou seja, fora aquilo que indicamos como exceção, todas as outras proteções continuam ativas e operando. A partir do momento que você abre o parâmetro sandbox com qualquer valor de exceção (ex: sandbox="allow-forms"), automaticamente todas as outras proteções são ativadas, bloqueando todo o resto.

Nota: Fora esses valores de exceção de segurança para sandbox listados aqui, existem outros... Ao abrir o parâmetro "sandbox", dentro das aspas, você pode usar o atalho "Ctrl + espaço" e abrir uma lista com os outros valores. Esses valores permitem conceder outros tipos de autorizações de segurança específicas para os seus iframes.