1. **что такое ASP.NET?**

ASP.NET – это часть технологии .NET, используемая для написания мощных клиент-серверных интернет приложений. Она позволяет создавать динамические страницы HTML. ASP.NET возникла в результате объединения более старой технологии ASP (активные серверные страницы) и .NET Framework. Она содержит множество готовых элементов управления, используя которые можно быстро создавать интерактивные web-сайты.

ASP.NET — это не продолжение ASP. Это концептуально новая технология Microsoft, созданная в рамках идеологии .NET. В ASP.NET заложено все, для того, чтобы сделать весь цикл разработки веб-приложения более быстрым, а поддержку более простой. ASP.NET основана на объектно-ориентированной технологии, но сохранила модель разработки asp: вы создаете программу и помещаете ее в директорию, выделенную сервером, и она будет работать. В ASP.NET появилось много новых функций, а существовавшие в asp значительно усовершенствованы.

В ASP.NET используются компилируемые языки. Во время компиляции проверяется синтаксическая корректность исходного текста. Скомпилированный в промежуточный язык код выполняется быстрее, и он будет таким же независимо от языка, который мы используем. Компилируемые языки поддерживают строгую типизацию.

Платформа .NET Framework предоставляет приложениям среду выполнения, сама непосредственно взаимодействуя с операционной системой. Выше лежит интерфейс ASP.NET приложений, на котором в свою очередь базируются веб-формы (ASP.NET страницы) и веб-сервисы. Интерфейс .NET Framework позволяет стандартизировать обращение к системным вызовам и предоставляет среду для более быстрой и удобной разработки. CLR обеспечивает единый набор сервисов для всех языков

**ASP.NET:** платформа для разработки web-приложений на основе платформы .NET Framework и работающих под управлением сервера IIS

**ASP.NET:** набор технологий для создания **web-приложений** и **web-сервисов** компании **Microsoft**. Последние версии: **ASP.NET 4.5 (VS 2017),** отдельные технологии имеют свою траекторию версий.

1. **что такое http-обработчик?**

ASP.NET: Http-handler - простейшее приложение ASP.NET

HTTP-обработчики (HTTP Handlers) используются для генерации содержимого ответа на HTTP-запрос. Мы можем использовать HTTP-обработчики для настройки обработчик запросов. При обработке одного запроса мы можем задействовать несколько различных модулей, но только один http-обработчик может быть сопоставлен с запросом.

HTTP-обработчик представляет класс, реализующий интерфейс System.Web.IHttpHandler. Данный интерфейс определяет один метод и одно свойство:

* ProcessRequest(context): данный метод в качестве параметра принимает объект контекста запроса HttpContext и генерирует ответ клиенту
* IsReusable: это свойство указывает, будет ли данный обработчик использоваться другими запросами

HTTP-обработчик выбирается системой после возникновения события MapRequestHandler. И сразу после выбора обработчика срабатывает событие PostMapRequestHandler

Непосредственная генерация ответа обработчиком происходит после события PreRequestHandlerExecute - после этого события происходит вызов метода ProcessRequest и генерация ответа. А сразу после генерации ответа возникает событие PostRequestHandlerExecute

Определение обработчика в файле web.config. Для этого в файле конфигурации в узле system.webServer (если его нет, то надо создать) пропишем определение обработчика

Все обработчики задаются в элементе handlers. Каждое определение имеет следующие атрибуты:

* name: уникальное название обработчика
* path: запрос URL, который будет обрабатываться обработчиком
* verb: тип запрос, например, GET. Мы также можем поставить любой тип запроса - в этом случае используется звездочка (\*)
* type: полный тип класса обработчика

1. **каким образом запросы маршрутизируются к http-обработчику?**

Определение обработчика в файле web.config. Для этого в файле конфигурации в узле system.webServer (если его нет, то надо создать) пропишем определение обработчика

Все обработчики задаются в элементе handlers. Каждое определение имеет следующие атрибуты:

* name: уникальное название обработчика
* path: запрос URL, который будет обрабатываться обработчиком
* verb: тип запрос, например, GET. Мы также можем поставить любой тип запроса - в этом случае используется звездочка (\*)
* type: полный тип класса обработчика