In this video, we will discuss the convention used by mvc to find views. Let's understand this with an example.

Notice that, the **Index**() action method does not specify the name of the view. So,   
**1.** How does asp.net mvc know, which view to use  
**2.** What locations does asp.net mvc search  
public class EmployeeController : Controller  
{  
    private EmployeeContext db = new EmployeeContext();  
  
    public ActionResult Index()  
    {  
        var employees = db.Employees.Include("Department");  
        return View(employees.ToList());  
    }  
  
    protected override void Dispose(bool disposing)  
    {  
        db.Dispose();  
        base.Dispose(disposing);  
    }  
}

To find the answer for the above 2 questions, delete the **"Index.cshtml"** view from **"Views/Employee"** folder. Run the application, and notice that we get an error message stating  
The view 'Index' or its master was not found or no view engine supports the searched locations. The following locations were searched:  
~/Views/Employee/Index.aspx  
~/Views/Employee/Index.ascx  
~/Views/Shared/Index.aspx  
~/Views/Shared/Index.ascx  
~/Views/Employee/Index.cshtml  
~/Views/Employee/Index.vbhtml  
~/Views/Shared/Index.cshtml  
~/Views/Shared/Index.vbhtml  
  
So, from the error message, it should be clear that, **MVC looks for a view with the same name as that of the controller action method** in the following locations  
**1.** Views/Shared  
**2.** Views/FolderNameMatchingControllerName  
  
**Please note that, the view extension can be any of the following**  
**a)**.cshtml  
**b)**.vbhtml  
**c)**.aspx  
**d)**.ascx  
  
**If I have all of the following files in "Views/Employee" folder, then MVC picks up "Index.aspx"**  
**a)** Index.aspx  
**b)** Index.cshtml  
**c)** Index.vbhtml  
**d)** Index.ascx  
  
**If you want to use "Index.cshtml" instead, then specify the full path as shown below.**  
public ActionResult Index()  
{  
    var employees = db.Employees.Include("Department");  
    return View("~/Views/Employee/Index.cshtml", employees.ToList());  
}  
  
If you specify only the name of the view along with it's extension as shown below, you will get an error.  
return View("Index.cshtml", employees.ToList());  
  
If you want to use a view name which is not inside the views folder of the current controller, then specify the full path as shown below.  
public ActionResult Index()  
{  
    var employees = db.Employees.Include("Department");  
    return View("~/Views/Home/Index.aspx", employees.ToList());  
}  
  
**Please note that asp.net mvc**  
**1.** Is all about convention over configuration  
**2.** Views folder contains one folder for each controller and a **"Shared"** folder  
**3.** Shared folder is used to store views shared between controllers. Master and Layout pages are stored in "Shared" folder.