

# Home Controller

---

```
public ActionResult Index()
{
    return View();
}
public ActionResult Contact()
{
    return View();
}
```

# Reservation Controller

---

```
private RailwayEntities db = new RailwayEntities();

// GET: Reservation
public ActionResult Index()
{
    var reservation = db.Reservation.Include(r => r.TrainDetail1).Include(r =>
r.TrainSchedule1);
    return View(reservation.ToList().Where(a => a.Name ==
Session["name"].ToString()));
}
public ActionResult Confirm()
{
    var reservation = db.Reservation.Include(r => r.TrainDetail1).Include(r =>
r.TrainSchedule1);
    return View(reservation.ToList().Where(a => a.Name ==
Session["name"].ToString()));
}

// GET: Reservation/Details/5
public ActionResult Details(int? id)
{
    if (id == null)
    {
        return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
    }
    Reservation reservation = db.Reservation.Find(id);
    if (reservation == null)
    {
        return HttpNotFound();
    }
    return View(reservation);
}

// GET: Reservation/Create
public ActionResult Create()
{
    ViewBag.TrainDetail = new SelectList(db.TrainDetail, "Id", "TrainName");
    ViewBag.TrainSchedule = new SelectList(db.TrainSchedule, "Id",
"Start_Station");
    return View();
}

// POST: Reservation/Create
// To protect from overposting attacks, please enable the specific properties you
want to bind to, for
// more details see https://go.microsoft.com/fwlink/?LinkId=317598.
[HttpPost]
[ValidateAntiForgeryToken]
```

```

        public ActionResult Create([Bind(Include =
        "Id,Journey_Date,Start_Station,End_Station,TrainDetail,TrainSchedule,Seats,Bill")]
        Reservation reservation)
        {
            if (ModelState.IsValid)
            {
                if (reservation.TrainDetail == 1)
                {
                    reservation.Bill = 1000 * reservation.Seats;
                }
                else if (reservation.TrainDetail == 2)
                {
                    reservation.Bill = 2000 * reservation.Seats;
                }
                reservation.Name = Session["name"].ToString();
                db.Reservation.Add(reservation);
                db.SaveChanges();
                return RedirectToAction("Confirm");
            }

            ViewBag.TrainDetail = new SelectList(db.TrainDetail, "Id", "TrainName",
            reservation.TrainDetail);
            ViewBag.TrainSchedule = new SelectList(db.TrainSchedule, "Id",
            "Start_Station", reservation.TrainSchedule);
            return View(reservation);
        }

        // GET: Reservation/Edit/5
        public ActionResult Edit(int? id)
        {
            if (id == null)
            {
                return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
            }
            Reservation reservation = db.Reservation.Find(id);
            if (reservation == null)
            {
                return HttpNotFound();
            }
            ViewBag.TrainDetail = new SelectList(db.TrainDetail, "Id", "TrainName",
            reservation.TrainDetail);
            ViewBag.TrainSchedule = new SelectList(db.TrainSchedule, "Id",
            "Start_Station", reservation.TrainSchedule);
            return View(reservation);
        }

        // POST: Reservation/Edit/5
        // To protect from overposting attacks, please enable the specific properties you
        want to bind to, for
        // more details see https://go.microsoft.com/fwlink/?LinkId=317598.
        [HttpPost]
        [ValidateAntiForgeryToken]
        public ActionResult Edit([Bind(Include =
        "Id,Journey_Date,Start_Station,End_Station,TrainDetail,TrainSchedule,Seats,Bill")]
        Reservation reservation)
        {
            if (ModelState.IsValid)
            {

```

```

        db.Entry(reservation).State = EntityState.Modified;
        db.SaveChanges();
        return RedirectToAction("Index");
    }
    ViewBag.TrainDetail = new SelectList(db.TrainDetail, "Id", "TrainName",
reservation.TrainDetail);
    ViewBag.TrainSchedule = new SelectList(db.TrainSchedule, "Id",
"Start_Station", reservation.TrainSchedule);
    return View(reservation);
}

// GET: Reservation/Delete/5
public ActionResult Delete(int? id)
{
    if (id == null)
    {
        return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
    }
    Reservation reservation = db.Reservation.Find(id);
    if (reservation == null)
    {
        return HttpNotFound();
    }
    return View(reservation);
}

// POST: Reservation/Delete/5
[HttpPost, ActionName("Delete")]
[ValidateAntiForgeryToken]
public ActionResult DeleteConfirmed(int id)
{
    Reservation reservation = db.Reservation.Find(id);
    db.Reservation.Remove(reservation);
    db.SaveChanges();
    return RedirectToAction("Index");
}

protected override void Dispose(bool disposing)
{
    if (disposing)
    {
        db.Dispose();
    }
    base.Dispose(disposing);}}}

```

# StamastersController

---

```
[Authorize]
public class stamastersController : Controller
{
    private RailwayEntities db = new RailwayEntities();

    // GET: stamasters
    public ActionResult Index()
    {
        return View(db.stamaster.ToList());
    }

    // GET: stamasters/Details/5
    public ActionResult Details(int? id)
    {
        if (id == null)
        {
            return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
        }
        stamaster stamaster = db.stamaster.Find(id);
        if (stamaster == null)
        {
            return HttpNotFound();
        }
        return View(stamaster);
    }

    // GET: stamasters/Create
    public ActionResult Create()
    {
        return View();
    }

    // POST: stamasters/Create
    // To protect from overposting attacks, please enable the specific properties you
    // want to bind to, for
    // more details see https://go.microsoft.com/fwlink/?LinkId=317598.
    [HttpPost]
    [ValidateAntiForgeryToken]
    public ActionResult Create([Bind(Include = "Id,Name,Code,DivisionName")]
    stamaster stamaster)
    {
        if (ModelState.IsValid)
        {
            db.stamaster.Add(stamaster);
            db.SaveChanges();
            return RedirectToAction("Index");
        }

        return View(stamaster);
    }
}
```

```

// GET: stamasters/Edit/5
public ActionResult Edit(int? id)
{
    if (id == null)
    {
        return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
    }
    stamaster stamaster = db.stamaster.Find(id);
    if (stamaster == null)
    {
        return HttpNotFound();
    }
    return View(stamaster);
}

// POST: stamasters/Edit/5
// To protect from overposting attacks, please enable the specific properties you
want to bind to, for
// more details see https://go.microsoft.com/fwlink/?LinkId=317598.
[HttpPost]
[ValidateAntiForgeryToken]
public ActionResult Edit([Bind(Include = "Id,Name,Code,DivisionName")] stamaster
stamaster)
{
    if (ModelState.IsValid)
    {
        db.Entry(stamaster).State = EntityState.Modified;
        db.SaveChanges();
        return RedirectToAction("Index");
    }
    return View(stamaster);
}

// GET: stamasters/Delete/5
public ActionResult Delete(int? id)
{
    if (id == null)
    {
        return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
    }
    stamaster stamaster = db.stamaster.Find(id);
    if (stamaster == null)
    {
        return HttpNotFound();
    }
    return View(stamaster);
}

// POST: stamasters/Delete/5
[HttpPost, ActionName("Delete")]
[ValidateAntiForgeryToken]
public ActionResult DeleteConfirmed(int id)
{
    stamaster stamaster = db.stamaster.Find(id);
    db.stamaster.Remove(stamaster);
    db.SaveChanges();
    return RedirectToAction("Index");
}

```

```
    }  
    protected override void Dispose(bool disposing)  
    {  
        if (disposing)  
        {  
            db.Dispose();  
        }  
        base.Dispose(disposing);  
    }  
}
```

# TrainDetailController

---

```
{  
    private RailwayEntities db = new RailwayEntities();  
  
    // GET: TrainDetail  
    public ActionResult Index()  
    {  
        return View(db.TrainDetail.ToList());  
    }  
    public ActionResult DayWiseTrain()  
    {  
        return View(db.TrainDetail.ToList());  
    }  
    [HttpPost]  
    public ActionResult DayWiseTrain(string abc)  
    {  
        if (string.IsNullOrEmpty(abc) == true)  
        {  
            return PartialView("SearchPartialView", db.TrainDetail.ToList());  
        }  
        else  
        {  
            var searchresult = db.TrainDetail.Where(a =>  
a.TrainName.Contains(abc)).ToList();  
            return PartialView("SearchPartialView", searchresult);  
        }  
        // return View(db.TrainDetail.ToList());  
    }  
  
    // GET: TrainDetail/Details/5  
    public ActionResult Details(int? id)  
    {  
        if (id == null)  
        {  
            return new HttpStatusCodeResult(HttpStatusCode.BadRequest);  
        }  
        TrainDetail trainDetail = db.TrainDetail.Find(id);  
        if (trainDetail == null)  
        {  
            return HttpNotFound();  
        }  
        return View(trainDetail);  
    }  
  
    // GET: TrainDetail/Create  
    public ActionResult Create()  
    {  
        return View();  
    }  
}
```



```

        // POST: TrainDetail/Create
        // To protect from overposting attacks, please enable the specific properties you
want to bind to, for
        // more details see https://go.microsoft.com/fwlink/?LinkId=317598.
        [HttpPost]
        [ValidateAntiForgeryToken]
        public ActionResult Create([Bind(Include =
        "Id,TrainName,Status,RouteId,No_of_compartment,C1AC,C2AC,C3AC,Sleeper,General")]
        TrainDetail trainDetail)
        {
            if (ModelState.IsValid)
            {
                db.TrainDetail.Add(trainDetail);
                db.SaveChanges();
                return RedirectToAction("Index");
            }

            return View(trainDetail);
        }

        // GET: TrainDetail/Edit/5
        public ActionResult Edit(int? id)
        {
            if (id == null)
            {
                return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
            }
            TrainDetail trainDetail = db.TrainDetail.Find(id);
            if (trainDetail == null)
            {
                return HttpNotFound();
            }
            return View(trainDetail);
        }

        // POST: TrainDetail/Edit/5
        // To protect from overposting attacks, please enable the specific properties you
want to bind to, for
        // more details see https://go.microsoft.com/fwlink/?LinkId=317598.
        [HttpPost]
        [ValidateAntiForgeryToken]
        public ActionResult Edit([Bind(Include =
        "Id,TrainName,Status,RouteId,No_of_compartment,C1AC,C2AC,C3AC,Sleeper,General")]
        TrainDetail trainDetail)
        {
            if (ModelState.IsValid)
            {
                db.Entry(trainDetail).State = EntityState.Modified;
                db.SaveChanges();
                return RedirectToAction("Index");
            }
            return View(trainDetail);
        }

        // GET: TrainDetail/Delete/5
        public ActionResult Delete(int? id)
        {
            if (id == null)

```

```

        {
            return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
        }
        TrainDetail trainDetail = db.TrainDetail.Find(id);
        if (trainDetail == null)
        {
            return HttpNotFound();
        }
        return View(trainDetail);
    }

    // POST: TrainDetail/Delete/5
    [HttpPost, ActionName("Delete")]
    [ValidateAntiForgeryToken]
    public ActionResult DeleteConfirmed(int id)
    {
        TrainDetail trainDetail = db.TrainDetail.Find(id);
        db.TrainDetail.Remove(trainDetail);
        db.SaveChanges();
        return RedirectToAction("Index");
    }

    protected override void Dispose(bool disposing)
    {
        if (disposing)
        {
            db.Dispose();
        }
        base.Dispose(disposing);
    }
}

```

# TrainScheduleController

---

```
private RailwayEntities db = new RailwayEntities();

// GET: TrainSchedule
public ActionResult Index()
{
    var trainSchedule = db.TrainSchedule.Include(t => t.TrainDetail);
    return View(trainSchedule.ToList());
}

// GET: TrainSchedule/Details/5
public ActionResult Details(int? id)
{
    if (id == null)
    {
        return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
    }
    TrainSchedule trainSchedule = db.TrainSchedule.Find(id);
    if (trainSchedule == null)
    {
        return HttpNotFound();
    }
    return View(trainSchedule);
}

// GET: TrainSchedule/Create
public ActionResult Create()
{
    ViewBag.Train_Id = new SelectList(db.TrainDetail, "Id", "TrainName");
    return View();
}

// POST: TrainSchedule/Create
// To protect from overposting attacks, please enable the specific properties you
want to bind to, for
// more details see https://go.microsoft.com/fwlink/?LinkId=317598.
[HttpPost]
[ValidateAntiForgeryToken]
public ActionResult Create([Bind(Include =
"Id,Start_Station,Destination_Station,Arrival_Time,Departure_Time,Distance_Between,Train_
Id")] TrainSchedule trainSchedule)
{
    if (ModelState.IsValid)
    {
        db.TrainSchedule.Add(trainSchedule);
        db.SaveChanges();
        return RedirectToAction("Index");
    }
}
```

```

        ViewBag.Train_Id = new SelectList(db.TrainDetail, "Id", "TrainName",
trainSchedule.Train_Id);
        return View(trainSchedule);
    }

    // GET: TrainSchedule/Edit/5
    public ActionResult Edit(int? id)
    {
        if (id == null)
        {
            return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
        }
        TrainSchedule trainSchedule = db.TrainSchedule.Find(id);
        if (trainSchedule == null)
        {
            return HttpNotFound();
        }
        ViewBag.Train_Id = new SelectList(db.TrainDetail, "Id", "TrainName",
trainSchedule.Train_Id);
        return View(trainSchedule);
    }

    // POST: TrainSchedule/Edit/5
    // To protect from overposting attacks, please enable the specific properties you
want to bind to, for
    // more details see https://go.microsoft.com/fwlink/?LinkId=317598.
    [HttpPost]
    [ValidateAntiForgeryToken]
    public ActionResult Edit([Bind(Include =
    "Id,Start_Station,Destination_Station,Arrival_Time,Departure_Time,Distance_Between,Train_
    Id")] TrainSchedule trainSchedule)
    {
        if (ModelState.IsValid)
        {
            db.Entry(trainSchedule).State = EntityState.Modified;
            db.SaveChanges();
            return RedirectToAction("Index");
        }
        ViewBag.Train_Id = new SelectList(db.TrainDetail, "Id", "TrainName",
trainSchedule.Train_Id);
        return View(trainSchedule);
    }

    // GET: TrainSchedule/Delete/5
    public ActionResult Delete(int? id)
    {
        if (id == null)
        {
            return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
        }
        TrainSchedule trainSchedule = db.TrainSchedule.Find(id);
        if (trainSchedule == null)
        {
            return HttpNotFound();
        }
        return View(trainSchedule);
    }
}

```

```
// POST: TrainSchedule/Delete/5
[HttpPost, ActionName("Delete")]
[ValidateAntiForgeryToken]
public ActionResult DeleteConfirmed(int id)
{
    TrainSchedule trainSchedule = db.TrainSchedule.Find(id);
    db.TrainSchedule.Remove(trainSchedule);
    db.SaveChanges();
    return RedirectToAction("Index");
}

protected override void Dispose(bool disposing)
{
    if (disposing)
    {
        db.Dispose();
    }
    base.Dispose(disposing);
}
}
```

# UserController

---

```
private RailwayEntities db = new RailwayEntities();

// GET: User
public ActionResult Index()
{
    return View(db.User.ToList());
}
public ActionResult Login()
{
    return View();
}
[HttpPost]
public ActionResult Login(User us, string returnUrl)
{
    var login = db.User.Where(a => a.Email == us.Email && a.Password ==
us.Password).FirstOrDefault();
    if (login != null)
    {
        FormsAuthentication.SetAuthCookie(login.Name, false);
        Session["name"] = login.Name;
        Session["Role"] = login.Role;
        if (login.Role == "Admin")
        {
            if (ReturnUrl != null)
            {
                return Redirect(ReturnUrl);
            }
            else
            {
                return RedirectToAction("Index", "User");
            }
        }
        else if (login.Role == "Employee")
        {
            if (ReturnUrl != null)
            {
                return Redirect(ReturnUrl);
            }
            else
            {
                return RedirectToAction("Index", "TrainDetail");
            }
        }
        else if (login.Role == "Customer")
        {
            if (ReturnUrl != null)
            {
                return Redirect(ReturnUrl);
            }
            else
            {
                return RedirectToAction("Index", "TrainDetail");
            }
        }
    }
    return RedirectToAction("Index", "User");
}
```

```

        {
            return RedirectToAction("Create", "Reservation");
        }
    }
}
else
{
    ViewBag.msg = "Invalid Credentials";
}
return View();
}

// GET: User/Details/5
public ActionResult Details(int? id)
{
    if (id == null)
    {
        return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
    }
    User user = db.User.Find(id);
    if (user == null)
    {
        return HttpNotFound();
    }
    return View(user);
}

// GET: User/Create
public ActionResult Create()
{
    return View();
}

// POST: User/Create
// To protect from overposting attacks, please enable the specific properties you
want to bind to, for
// more details see https://go.microsoft.com/fwlink/?LinkId=317598.
[HttpPost]
[ValidateAntiForgeryToken]
public ActionResult Create([Bind(Include = "Id,Name,Email,Password,Role")] User
user)
{
    if (ModelState.IsValid)
    {
        db.User.Add(user);
        db.SaveChanges();
        return RedirectToAction("Login");
    }
    else
    {
        ViewBag.msg = "Invalid Credentials";
    }

    return View(user);
}

// GET: User/Edit/5
public ActionResult Edit(int? id)

```

```

    {
        if (id == null)
        {
            return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
        }
        User user = db.User.Find(id);
        if (user == null)
        {
            return HttpNotFound();
        }
        return View(user);
    }

    // POST: User/Edit/5
    // To protect from overposting attacks, please enable the specific properties you
    want to bind to, for
    // more details see https://go.microsoft.com/fwlink/?LinkId=317598.
    [HttpPost]
    [ValidateAntiForgeryToken]
    public ActionResult Edit([Bind(Include = "Id,Name,Email,Password,Role")] User
user)
    {
        if (ModelState.IsValid)
        {
            db.Entry(user).State = EntityState.Modified;
            db.SaveChanges();
            return RedirectToAction("Index");
        }
        return View(user);
    }
    public ActionResult Logout()
    {
        FormsAuthentication.SignOut();
        Session.Abandon();
        return RedirectToAction("Login", "User");
    }

    // GET: User/Delete/5
    public ActionResult Delete(int? id)
    {
        if (id == null)
        {
            return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
        }
        User user = db.User.Find(id);
        if (user == null)
        {
            return HttpNotFound();
        }
        return View(user);
    }

    // POST: User/Delete/5
    [HttpPost, ActionName("Delete")]
    [ValidateAntiForgeryToken]
    public ActionResult DeleteConfirmed(int id)
    {
        User user = db.User.Find(id);

```



```
        db.User.Remove(user);
        db.SaveChanges();
        return RedirectToAction("Index");
    }

    protected override void Dispose(bool disposing)
    {
        if (disposing)
        {
            db.Dispose();
        }
        base.Dispose(disposing);
    }
}
```