namespace WebApplication7.Controllers

{

[Route("api/[controller]")]

[ApiController]

public class StudentController : ControllerBase

{

//SqlConnection con = new SqlConnection("Data Source=DESKTOP-27TFO05;Initial Catalog=Users;Integrated Security=True");

List<Student> st = new List<Student>()

{

new Student{id=1,name="Amjad",department="BSE"},

new Student{id=2,name="Ali",department="CS"},

new Student{id=3,name="Haider",department="BSE"},

new Student{id=4,name="Haris",department="CS"},

};

[HttpGet]

public IActionResult GetAllStudent()

{

//if (st.Count == 0)

//{

// return NotFound("No student found");

//}

List<UserDetail> users = new List<UserDetail>();

using(SqlConnection con = new SqlConnection("Data Source=DESKTOP-27TFO05;Initial Catalog=UserData;Integrated Security=True"))

{

con.Open();

SqlCommand cmd = new SqlCommand("Select \* from Users",con);

SqlDataReader reader = cmd.ExecuteReader();

while (reader.Read())

{

UserDetail user = new UserDetail();

user.Id = (int )reader["ID"];

user.Firstname = reader["FirstName"].ToString();

user.Lastname = reader["LastName"].ToString();

user.Email = reader["Email"].ToString();

user.Username = reader["Username"].ToString();

user.Password = reader["Password"].ToString();

users.Add(user);

}

}

return Ok(users);

}

[HttpGet("GetStudent")]

public IActionResult Get(int id)

{

//var s = st.SingleOrDefault(x => x.id == id);

//if (s == null)

//{

// return NotFound("No student found");

//}

UserDetail user = new UserDetail();

using (SqlConnection con = new SqlConnection("Data Source=DESKTOP-27TFO05;Initial Catalog=UserData;Integrated Security=True"))

{

con.Open();

SqlCommand cmd = new SqlCommand("Select \* from Users where ID="+id, con);

SqlDataReader reader = cmd.ExecuteReader();

if (reader.Read())

{

user.Id = (int)reader["ID"];

user.Firstname = reader["FirstName"].ToString();

user.Lastname = reader["LastName"].ToString();

user.Email = reader["Email"].ToString();

user.Username = reader["Username"].ToString();

user.Password = reader["Password"].ToString();

}

else

{

user = null;

return Ok("Not found");

}

}

return Ok(user);

}

[HttpPost]

public IActionResult AddStudent(Student s)

{

st.Add(s);

if (st.Count == 0)

{

return NotFound("No student found");

}

return Ok(st);

}

[HttpPut("{id}")]

public IActionResult UpdateStudent(int id, UserDetail user)

{

//var student = st.FirstOrDefault(x => x.id == id);

//if (student == null)

//{

// return NotFound("Student not found");

//}

//student.name = s.name;

//student.department = s.department;

using (SqlConnection con = new SqlConnection("Data Source=DESKTOP-27TFO05;Initial Catalog=UserData;Integrated Security=True"))

{

con.Open();

SqlCommand cmd = new SqlCommand("Select \* from Users where ID=" + id, con);

SqlDataReader reader = cmd.ExecuteReader();

if (reader.Read())

{

string sql = "UPDATE Users SET Firstname=@Firstname, Lastname=@Lastname, Email=@Email, Username=@Username, Password=@Password WHERE Id=@Id";

SqlCommand cmd1 = new SqlCommand(sql,con);

cmd1.Parameters.AddWithValue("@Id", user.Id);

cmd1.Parameters.AddWithValue("@Firstname", user.Firstname);

cmd1.Parameters.AddWithValue("@Lastname", user.Lastname);

cmd1.Parameters.AddWithValue("@Email", user.Email);

cmd1.Parameters.AddWithValue("@Username", user.Username);

cmd1.Parameters.AddWithValue("@Password", user.Password);

reader.Close();

cmd1.ExecuteNonQuery();

}

else

{

return Ok("Not found");

}

}

//List<UserDetail> users = new List<UserDetail>();

//using (SqlConnection con = new SqlConnection("Data Source=DESKTOP-27TFO05;Initial Catalog=UserData;Integrated Security=True"))

//{

// con.Open();

// SqlCommand cmd = new SqlCommand("Select \* from Users", con);

// SqlDataReader reader = cmd.ExecuteReader();

// while (reader.Read())

// {

// UserDetail user1 = new UserDetail();

// user1.Id = (int)reader["ID"];

// user1.Firstname = reader["FirstName"].ToString();

// user1.Lastname = reader["LastName"].ToString();

// user1.Email = reader["Email"].ToString();

// user1.Username = reader["Username"].ToString();

// user1.Password = reader["Password"].ToString();

// users.Add(user1);

// }

//}

return Ok(user);

}

[HttpDelete]

public IActionResult DeleteStudent(int id)

{

var s = st.SingleOrDefault(x => x.id == id);

if (s == null)

{

return NotFound("No student found");

}

st.Remove(s);

if (st.Count == 0)

{

return NotFound("No student found");

}

return Ok(st);

}

}

}