**COURSEWORK SUBMISSION FORM**

| **STUDENT USE** | | **STAFF USE** | |
| --- | --- | --- | --- |
| Module Name | Web Applications Development | First Marker’s  (acts as signature) |  |
| Module Code | 5COSC017C | Second Marker’s  (acts as signature) |  |
| Lecturer Name | Isomiddin Abdunabiev | Agreed Mark |  |
| UoW Student IDs | 18150193/1 | **For Registrar’s office use only (hard copy submission)** | |
| WIUT Student IDs | 00010989 |
| Deadline date | 28/02/2023 |
| Assignment Type | 🗌Group🗌Individual |

**SUBMISSION INSTRUCTIONS**

**COURSEWORKS *must* be submitted in *both* HARD COPY (to the Registrar’s Office) *and* ELECTRONIC unless instructed otherwise.**

For hardcopy submission instructions refer to: <http://intranet.wiut.uz/Shared%20Documents/Forms/AllItems.aspx> - Coursework hard copy submission instructions.doc

For online submission instructions refer to: <http://intranet.wiut.uz/Shared%20Documents/Forms/AllItems.aspx> - Coursework online submission instructions.doc

| **MARKERS FEEDBACK (Continued on the next page)** |
| --- |
|  |

**SRP code example**

I will apply this principle by splitting the functionality of one class in to two classes if I`ll have 2 reasons to change for this class

**Without**

public class Order {

public void checkOrderIntegrity(){

*/\*Some Logic\*/*

}

public void The\_Order\_Shipping\_Cost(){

*/\*Some Logic\*/*

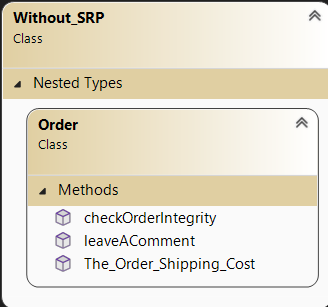
}

public void leaveAComment() {

*/\*Some Logic\*/*

}

}



**With**

public class IntegrityOfOrderChecking{

public void checkOrderIntegrity(){

*/\*Some Logic\*/*

}

}

public class OrderShippingCost{

public void count\_The\_Order\_Shipping\_Cost(){

*/\*Some Logic\*/*

}

}

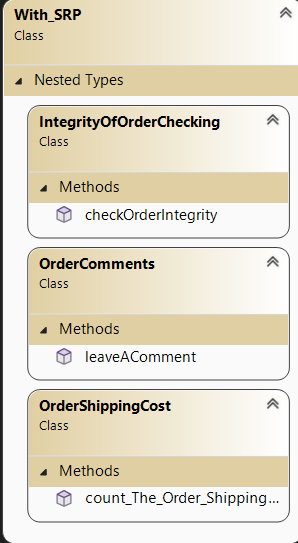
public class OrderComments{

public void leaveAComment() {

*/\*Some Logic\*/*

}

}



**DRY code example**

I will apply the DRY Principle in my application in case if I'll have some piece of logic represented twice. I`ll remove this piece of logic and re-write the code immediately

**Without**

public class Order

{

public string Description{ get; set; }

public override string ToString()

{

return Description;

}

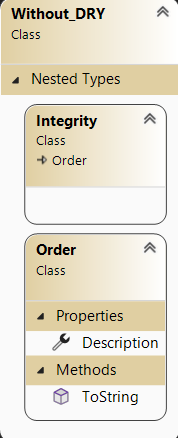
}

public class Integrity : Order

{

*/\*Some Logic\*/*

}



**With**

namespace Delivery

{

public class Integrity

{

public DateTime DeliveryTime { get; set; }

public int Price{ get; set; }

public string OrderDescription { get; set; }

public void SendOrderToCourier(TaxiDriver taxidriver)

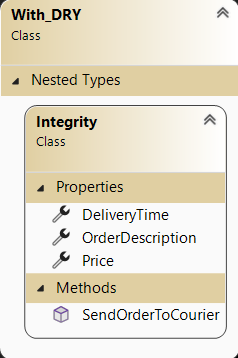
{

*/\*Some Logic\*/*

}

}

}



Link to GitHub:

<https://github.com/JSDeveloper545/WAD-00010989.git>