* שדה בוליאני לרשימות ריקות / מלאות
* תמונות – לקחת מ-images
* להוריד USER מ-BO?
* inner exception
* לבדוק חלונות עבור רשימות ריקות
* מספר רץ – מחלקה או XML?
* בכל get ב-PL – לטפל במקרה שריק
* Get עם תנאי – יש דרך לקרוא ל-DL?
* PO? שיבוט ב-BL?
* Singelton ב-BL?
* תחנות עוקבות: חישובים יכולים להיות רק עבור הנתונים הראשוניים (שמכניסים לdal), בשאר המקומות צריך לעשות קלט. לכן לא ניתן לוותר על היישות הזו.

*האוטובוס יסע בתהליכונים מסוימים - כלומר תהיה נסיעה של האוטובוס כיישות של BO, וכל פעם שתותחל נסיעה לפי לוח זמנים יווצר תהליכון. כמו כן יהיה אלמנט של רנדומנילזציה, כלומר הזמנים לא יהיו מדוייקים לפי לוח הזמנים, אלא בטווח אקראי כלשהו. לכן נצטרך כל הזמן לבדוק איפה הוא בפועל כדי לחשב נסיעה בין תחנות. הנסיעה תהיה סוג של ארוע -לפי תבנית המשקיף ( observer) וכל מי שרוצה לקבל ממנה מידע כדי לדעת את הנסיעה, ירשם לאירוע. א. צריך גם את יישות לוח הזמנים בשכבת הDO. אבל רק אחרי שכל שאר הפרויקט עשוי. ב. תהיה יישות נסיעה של אוטובוס, ושאילתא של נסיעת נוסע מתחנה לתחנה - בשכבת הBL.*

*נסיעת משתמש: מישהו נמצא בתחנה מסוימת הוא יודע שם אילו אוטובוסים עוברים, מחפש ביניהם רק את האוטובוסים שיש לו את התחנת ירידה איפה שהוא רוצה ויבדוק מה הכי מהר. זו שאילתא שמבוצעת ב-BL על סמך כל הנתונים. לא אמור להיות ב-DO.*

*אם רוצים ישות משתמש – בונוס – אז שומרים את היסטוריית כל הבקשות שלו.*

*אוטובוס בנסיעה: הישות BusOnTrip עוברת ל-BO  ויורד ממנה השדה שמזהה אוטובוס.*

*שינוי בקובץ config*

בסוף:

* בונוסים אפשריים: אוטובוס + סימון למחיקה (לא כולל תחנות עוקבות) + שיבוט + הצפנה
* לתעד בונוסים
* לתעד תוכנית
* לעבור על ההוראות של התרגיל
* למחוק קבצים מיותרים

if (lineStation.PathIndex == lineStations.Count())

{

DO.Line line = dal.getLine(lineStation.NumberLine);

line.LastStation = lineStations.ElementAt(lineStations.Count() - 1).ID;

}

if (lineStation.PathIndex == 1)

{

DO.Line line = dal.getLine(lineStation.NumberLine);

line.FirstStation = lineStations.First().ID;

}

<Window.Resources>

<DataTemplate x:Key="LineStations">

<Grid>

<Grid.ColumnDefinitions>

<ColumnDefinition Width="\*"/>

<ColumnDefinition Width="\*"/>

</Grid.ColumnDefinitions>

<Label x:Name="ID" Grid.Column="0" Content="{Binding Path=ID}"/>

<Label x:Name="Name" Grid.Column="1" Content="{Binding}"/>

</Grid>

</DataTemplate>

</Window.Resources>