בסיס הנתונים היחסי מכיל טבלה אחת שמכילה פרטי סטודנטים לפי התכונות הבאות:

ת.ז (מפתח), שם פרטי, שם משפחה, כתובת (עיר מגורים, רחוב, מספר בית, מיקוד), תאריך לידה (שנה, חודש, יום), שנת התחלת הלימודים, מחלקה, כמות נ.ז שנצברו עד כה, ממוצע הציונים, כמות הנכשלים, מיקום הסטודנט לפי ציוניו המיקום ייקבע ביחס לסטודנטים במחלקה של הסטודנט, ידורגו רק סטודנטים עם כמות נ"ז שאינה נופלת מחציון הנ"ז של המחלקה, לאלו שאינם עומדים בקריטריון תהיה בשדה הזה ההודעה: אינו מדורג (כלומר זה שדה מחושב מחדש עם כול עדכון בסיס הנתונים).

משימות:  
  
בניית בסיס הנתונים (mySql) המכיל את הטבלה האחת כפי שתוארה. הטבלה צריכה להכיל לפחות 50 רשומות, ניתן לחולל אותן אקראי או לקודד את הנתונים. הנתונים ההתחלתיים ייכנסו לבסיס הנתונים דרך אירוע שיבנה את בסיס הנתונים. כול פעם שהאירוע הזה יופעל יימחקו כול הטבלאות וייבנו מחדש למצבן המקורי.

בנוסף לבניית האוטומטית הראשונית יש לכתוב ממשק עדכון השורות בטבלאות. יש לאפשר להוסיף עוד שורות, יש לאפשר לבטל שורות ולעדכן נתונים בשורות כולל התמונות. כל זה בממשק נוח, ידידותי וגרפי. המשתמש אינו מזין בשום פעולת עדכון שאילתות התחביר SQL . יש להשתמש היכן שניתן בתיבות בחירה מתוך רשימות מוכנות מראש (כמו שמות ערים, רחובות, מאגר תמונות). סטודנטים חדשים, או גריעת סטודנטים יחייבו עדכון דרך התוכנה של שדה מיקום הסטודנט.

יש לכתוב ממשק שאילתות. ליד כול שדה בטבלה ייפתח חלון שיאפשר לבחור את פעולת ההשוואה הנדרשת והמשתמש יזין את הערך מולו תושווה התכונה, וכן תתאפשר בחירת הערכים מולם משווים גם מתוך תיבות בחירה.

יש לאפשר הצגת הטבלה והצגת כול פלט של שאילתה כך שניתן למיין את השורות לפי אחת התכונות. יש לאפשר שתי רמות מיון, למשל מיון ראשי לפי מחלקה ובתוכו לפי ממוצע ציונים, או לפי מקום הסטודנט. כול מיון יתאפשר בסדר עולה או יורד.

יש לאפשר הוספת עמודה לטבלה, למשל מין הסטודנט עם ערך ברירת מחדל (אח"כ כמו כול שורה בטבלה ניתן לשנות את הנתון). יש לאפשר גריעת עמודה מהטבלה (למעט עמודת המפתח כמובן). יש לטפל בהשלכות הנובעות מכך. למשל אם תבוטל עמודת ציון כול הנתונים בעמודת דירוג סטודנט יהפכו לבלתי מדורג.

יצירתיות מתבקשת בכול הקשור לממשק המשתמש.

הממשק יפותח ב-javaFx, והמחלקה ב-java8 u-20 , סביבת עבודה luna eclipse.

דרישה סף מהותית והכרחית במערכת היא הפרדה בין שרת ללקוח ועבודה במולטיטרדינג. ברמה הכללית השרת מבצע את ה- SQL (עדכון ואחזור), את חישוב המיקום של הסטודנט, ומחלקות הלקוח יוצרות את הממשקים ומקבלות את האירועים, בעקבותיהם הם מעבירים לשרת את ה-SQL המתאים ומקבלות ממנו את טבלת הפלט, אותה הן מציגות. השרת הוא tread עצמאי ומתייחס לכול לקוח כ-thread נפרד.