**שלב 6 - הוספת צבע ו"חומר" לגופים, הוספת מקורות אור, מימוש מודל פונג פשוט**

לפני שממשכים לשלב הזה יש לוודא שלא שכחתם ליצור **tag** ב-**commit** האחרון עבור השלב הקודם.

***נ.ב.1*** *אין לבנות בנאי ברירת מחדל, בנאי העתקה, והעמסת* ***equals*** *במחלקות של השלב הזה, אלא אם נאמר אחרת במפורש.*

***נ.ב.2*** *חובה להקפיד על תיעוד בפורמט* ***javadoc*** *לכל המחלקות והפעולות החדשות ולעבוד ע"פ הסדר כפי שנלמד בקורס!*

***נ.ב.3*** *השלב ניתן לשבועיים* ***אך שימו לב****: השלב כולל לא מעט חשיבה, הבנה ועבודה – לא כדאי להתעכב אתו אלא להתחילו ולהתקדם כמה שיותר מהר!*

**הכנות**

קודם כל ניקח מתיבת ההגשה מודולים של בדיקות לכל החלקים של השלב ונשים אותם בחבילת **unitests**: בדיקה בסיסית עם צבעי גופים, בדיקות מקורות תאורה עם זוג משולשים ועם ספרה.

**שלב א' – הוספת צבע לגופים גאומטריים**

1. נהפוך את האינטרפייס **Geometry** למחלקה אבסטרקטית:
   1. נשנה **interface** ל-**abstract class** ונשנה את הקשר לממשק **Intersectable** מ-**extends** ל-**implements**
   2. נוסיף להצהרה של הפעולה **getNormal** הרשאה **public** ומילת מפתח **abstract**
   3. נוסיף בתחילת המחלקה שדה בשם **emmission** מסוג **Color** עם הרשאה **protected** ועם אתחולו לצבע שחור – **Color.BLACK**
   4. נוסיף פעולת מאחזרת (גטר) עבור השדה החדש – כמובן עם תעוד מתאים בפורמט **javadoc**
   5. נוסיף פעולת עדכון (סטר) עבור השדה החדש שתחזיר את האובייקט עצמו (כמו בתבנית **Builder**) – כמובן עם תעוד מתאים בפורמט **javadoc**
   6. נשנה בכל המחלקות שהצהירו על מימוש הממשק (**Plane**, **Polygon**, **Tube**) מ-**implements** ל-**extends**
2. נעדכן את המודול **RenderTests.java** עם הקובץ מתיבת ההגשה או נעביר את מתודת הבדיקה הנוספת **basicRenderMultiColorTest** משם אל המודול שלכם.
3. נוסיף מחלקת עזר פנימית סטטית (כמבנה נתונים פסיבי לגמרי - **PDS**) בשם **GeoPoint** בתוך ממשק **Intersectable**:
   1. נוסיף את המחלקה בצורה הבאה:

**public** **static** **class** GeoPoint {

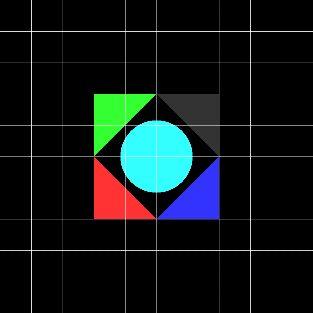
**public** Geometry geometry;

**public** Point3D point;

}

* 1. נוסיף לפני המחלקה הזו תיעור בפורמט **javadoc**
  2. נוסיף בתוך המחלקה בנאי המקבל שני פרמטרים עובר אתחול של שני השדות, וכמובן נוסיף תיעוד לבנאי בפורמט **javadoc**
  3. נוסיף למחלקה העמסת המתודה **equals** לפי הכללים כמו בשלבים הראשונים במחלקות בחבילה **primitives**, שתבדוק שההשואה מתבצעת על אותו גוף גאומטרי והנקודות זהות
  4. נוסיף בממשק מתודה **findGeoIntersections** המקבלת פרמטר של קרן ומחזירה **List<GeoPoint>**
  5. נוסיף בהתאם את המימושים של המתודה החדשה במחלקות הממשות את הממשק למימוש של המתודה החדשה – גם **Geometries** וגם במחלקות הגאומטריות הבסיסיות (בהן בשדה הגאומטריה באוביקט המוחזר יושם ערך **this**) בהתאם (שימו לב שזה לא בדיוק העתק\הדבק)
  6. שימו לב: במחלקות המצולע (הפוליגון) והמשולש חובה לעדכן את השדה **geometry** באובייקט שהתקבל במקור מהמישור שקשור למצולע\המשולש לערך **this**
  7. נוסיף במחלקה Ray מתודה **getfindClosestGeoPoint** (עם תיעוד ומימוש בהתאם), נוסיפף בתחילת המודול את היבוא הבא:

**import** geometries.Intersectable.GeoPoint;

* 1.  נוסיף מתודת בדיקה למתודה החדשה הזו
  2. נשנה בהתאם את המתודות והקוד שלהן: **~~renderImage~~**~~,~~ **traceRay**, ו-**calcColor**, נוסיף בתחילת המודול את היבוא הבא:

**import** geometries.Intersectable.GeoPoint;

* 1. נעדכן את הפונקציה **calcColor** כך שתוסיף את הצבע של האובייקט לצבע הנקודה
  2. נריץ את **כל** הטסטים ונוודא את הצלחתם, כמובן שימו לב על קובץ התמונה החדש שנוצר ונבדוק את תקינותו:
  3. ארגון קוד מחדש (**refactoring**): נשים לב על כפילות הקוד של החיתוכים ולכן נבצע את הפעולות הבאות:
     1. נחליף הצהרת למתודה הישנה **findIntersections** בממשק **Intersectable** להגדרה הבאה (הכוללת מימוש בררת מחדל עבור המתודה שבממשק):

**default** List<Point3D> findIntersections(Ray ray) {

**var** geoList = findGeomIntersections(ray);

**return** geoList == **null** ? **null**

: geoList.stream().map(gp -> gp.point).collect(Collectors.*toList*());

}

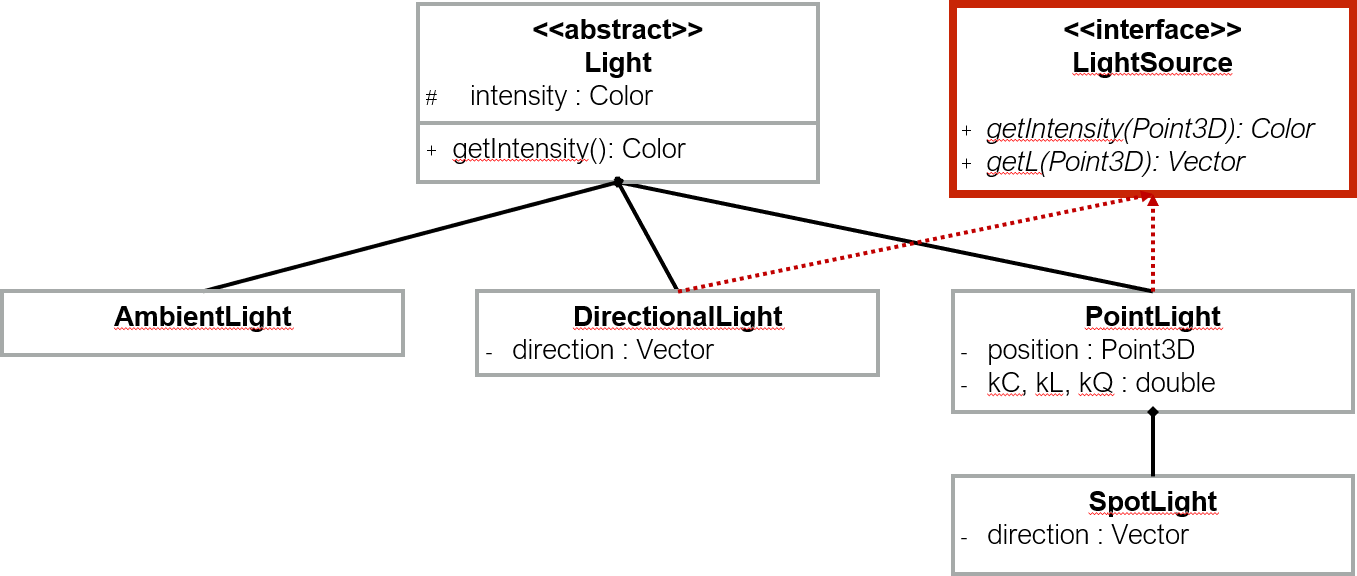
* + 1. נמחק את כל המימושים של הפונקציה הישנה הנ"ל מכל מחלקות הגופים הגאומטריים – הפשוטים והמורכב
    2. נריץ מחדש את כל הטסטים ונבדוק את תקינותם

**שלב ב' – הוספת מקורות אור**

1. בחבילה **elements** נוסיף מחלקה אבסטרקטית חדשה בשם **Light** ללא הרשאה
   1. נוסיף למחלקה תיעוד בפורמט **javadoc**
   2. במחלקה נוסיף שדה בשם **intensity** מסוג **Color** עם הרשאה פרטית
   3. נוסיף במחלקה בנאי (עם הרשאה **protected**) המאתחל את השדה ומתודה מאחזרת (גטר) עבור השדה (לא נוסיף מתודה מעדכנת עבור השדה!) – עם תיעוד בפורמט **javadoc**
2. נשנה את הגדרת מחלקת **AmbientLight** כך שתירש מהמחלקה החדשה **Light**, נמחק מהמחלקה את השדה **intensity** ומתודה מאחזרת שלו, נשנה את הבנאי כך שיפעיל את בנאי האב עם עוצמה סופית של תאורה סביבתית () ונעדכן את התעוד שבמחלקה אם נדרש, כמו כן נעדכן בהתאם את בנאי ברירת המחדל כך שיעביר צבע שחור לבנאי האב.
3. בחבילה **elements** נוסיף ממשק בשם **LightSource** עם הרשאה **public**
   1. נוסיף לממשק תיעוד בפורמט **javadoc**
   2. נוסיף בממשק הגדרת שתי הפונקציות הבאות, נוסיף להן תיעוד בפורמט **javadoc**

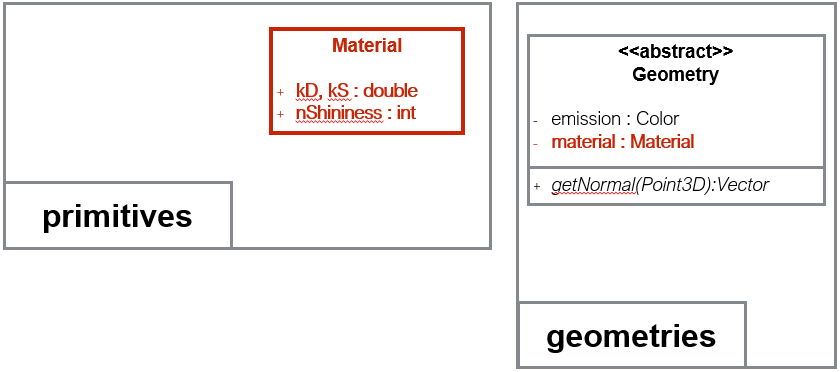
**public** Color getIntensity(Point3D p);

**public** Vector getL(Point3D p);

1. בחבילה **elements** נוסיף מחלקות **DirectionalLight** ו-**PointLight** היורשות מ-**Light** וממשות את **LightSource**, כמו כן **SpotLight** שיורשת מ-**PointLight**
   1. במחלקות החדשות האלה נגדיר את השדות בהתאם לארכיטקטורה:
   2. השדות במחלקות החדשות עם הרשאה **private**
   3. לא נגדיר מתודות מאחזרות לשדות האלה כלל
   4. בתאורה נקודתית (**PointLight**), עבור מקדמי ההנחתה נגדיר סטרים משרשרים בדומה לסטרים בתבנית **Builder**
   5. לכל המחלקות נוסיף בנאי שיקבל בפרמטרים את הערכים לאתחול כל השדות צבע, מיקום וכיוון (כל מחלקה לפי מה שהיא צריכה)

**נ.ב.** אתחול ברירת המחדל עבור מקדמי ההנחתה:

* 1. לכל הבנאים נוסיף תיעוד בפורמט **javadoc**
  2. בכל המחלקות נממש את פונקציות הממשק הנ"ל ע"פ מה שנלמד בקורס ובמעבדה
  3. **תזכורת** – מימוש **getL** בתאורה כיוונית יחזיר את ערך כיוון התאורה
  4. במימושים חובה להקפיד על עקרונות עיצוב, כולל **DRY**

1. במחלקה **Scene** נוסיף שדה **lights** מסוג **List<LightSource>**
   1. השדה יאותחל עם **new LinkedList<LightSource>()**
   2. לשדה נוסיף מתודה מעדכנת המחזירה את הסצנה – בהתאם לתבנית **Builder** (כמובן עם תיעוד בפורמט **javadoc**)
2. בחבילה **primitives** נוסיף מחלקה **Material** עם השדות בהרשאה ציבורית כפי שהוגדר בקורס (המחלקה תהיה **PDS** עם תבנית **Builder** בסיסית – פעולות מעדכנות המחזירות את האובייקט לצורך שרשור):
   1. אתחול ברירת מחדל של כל השדות – לערך 0.
   2. נוסיף למחלקה פעולות מעדכנות (סטרים) לכל השדות כנ"ל – עם תיעוד בפורמט **javadoc**
3. נוסיף במחלקה **Geometry** שדה בשם **material** מסוג **Material** הנ"ל עם הרשאה פרטית
   1. אתחול בררת מחדת של הדה יהיה עם חומר ברירת מחדל **new Material()**
   2. נוסיף פעולת מאחזרת (גטר) עבור השדה החדש – כמובן עם תיעוד מתאים בפורמט **javadoc**
   3. נוסיף עבור השדה פעולה ~~מאחזרת~~ מעדכנת (סטר) בפורמט של תבנית **Builder** (שתחזיר את הגאומטריה עצמה)
4. בפונקציה **calcColor** נוסיף את הקוד המוסיף את השפעת מקורות האור על הנקודה שעבור מחושב הצבע ע"פ מודל פונג הפשוט – ע"פ הנלמד בקורס ובמעבדה
   1. נממש את **כל פונקציות העזר** שהשתמשנו ונוסיף להן תיעוד
5. נוסיף עכשיו טסט של מקורות אור שקיבלנו בתיבת ההגשה עבור בדיקות מודל פונג, נריץ את **כל** הטסטים (הישנים והחדשים) ונתבונן בתמונות שנוצרו, אם נגלה תופעות לא רצויות – נדבג ונתקן את הקוד שכתבנו
6. **במודול הטסטים של תאורה נוסיף שתי פעולות בדיקה עבור מקורות אור מרובים** (הכוללים את כל סוגי התאורה) – פעולה אחת עבור הכדור ופעולה השניה עבור זוג המשולשים תוך שינוי. עליכם לשנות את מיקומי מקורות האור מהטסטים המקוריים כך שתוכר השפעה משותפת של כל מקורות האור, כמו כן אליכם "לשחק" עם המקדמים ועוצמות על מנת לקבל תמונה משכנעת ומרהיבה הן עבור כדור והן עבור זוג המשולשים. אין לשנות את הפרמטרים של הכדור ושל המשולשים.
7. בשלב של **refactoring** – במתודות עזר של סריקת קרן (במתודות הקשורות לחישוב מודל פונג) עליכם להפעיל את ההיגיון הבריא ולעשות את מה שאפשרי על מנת לייעל את עבודתן - גם אם צריך להוסיף בשביל זה פרמטרים בהשוואה למה שמופיע במצגת!

**שלב לבונוס**

ניתן לקבל בונוס של עד 2 נק' ע"י (1 נק' עבור כל אחת מהסעיפים):

* קבלת מקור תאורה ספוט עם אלומת עור צרה יותר ("צרות" האלומה מתכווננת ע"י פרמטר נוסף). רמז – הסתכלו על "צרות" ההבזק (קומפוננטה ספקולרית), הסעיף כולל שני טסטים של ספוט משופר – עבור הכדור ועבור זוג המשולשים כנ"ל.

**נ.ב.** במודול הטסטים אין בדיקות עבור הבונוס – עליכם להוסיף לפחות שתי בדיקות – עבור הספרה ועבור המשולשים – להמחשת העבודה שעשיתם בבונוס.

* השלמת השלב תוך שבוע אחד במקום שבועיים.

ראו נספח של בעיות נפוצות בעמוד הבא

בהצלחה!

**בעיות נפוצות**

|  |  |
| --- | --- |
| בעיה | סיבות אפשריות |
| שלב 1: אין צבע (כל הצורות אפורות) | לה הצבתם צבע מעודכן בחזר למשתנה של צבע  **color =** color.add(gp.geometry.getEmmission()); |
| שלב 1: סידור צבעי המשולשים שונה מהדוגמה במטלה | בשני השלבים הקודם לא התנהלתם נכון עם האינדקסים של פיקסלים |
| שלב 2: עוצמת התאורה רבה מידי באחד או כל הטסטים | אפשרות 1. וקטור המוחזר מאחד או כל המימושים של getL איינו מנורמל  אפשרות 2. שימוש בפעולת reduce של Color במקום פעולה scale  אפשרות 3. באגים נוספים בקוד (יופיעו כאן ברגע שנפרוש ונכיר אותם) |
| שלב 2: בטסט ספוט מעל משולשים יש מעין פרפר של תאורה כאילו הספוט מאיר לשני הכיוונים | לא התבצעה בדיקה ששכיוון הקרן (של מצלמה) לנקודה וכיוון ממקור האור לנקודה – זהים בסימן הכפלתם הסקלרית לנורמל בנקודה הזו. |