שאלה 4   
  
א  
mutable החשיבות שלו הוא התכונה העיקרית שלו והיא שהוא ניתן לשינוי  
ולכן מאפשר לנו לבצע שינויים בנתונים ולהתנהל בצורה דינמית ביחס לערכים לוקליים (נוכל לשנות אותם)  
  
immutable החשיבות שלו הוא התכונה העיקרית שלו והיא שהוא לא ניתן לשינוי   
ולכן מאפשר לנו ליצור דברים קבועים בלתי ניתנים לשינוי כגון (tuple)ולכן יותר בטיחות לדוגמה לא נוכל לשנות מידע לאחר היצירה שלו   
  
  
ב  
מרבית השפות מעדיפות את המודל הלקסיקלי(מצביעים לסביבת ההגדרה ) על פני הדינמי (מצביעים לסביבת ההפעלה) מכיוון שלפונקציות יש גישה למשתנים בסביבת ההגדרה שלהן ולכן זה יוצר תבנית ברורה של סדר פעולות וביצוע קשירות בין משתנים לערכים ולכן נוכל להבין קוד בלי להריץ אותו רק בעזרת מוגדל הסביבות הלקסיקלי   
  
  
  
ג  
Applicativ order הוא עיקרון שנובע מהרצון ליעילות מתכנת רבה יותר צורת הפעולה שלו היא הערכת כל הארגומנטים ואז קריאה לאופרטורים לכן תמיד נעבור על כל הקוד בניגוד ל"נורמל אורדר" וזה מבטיח למצוא קטעים שלא עובדים או שגויים בקוד   
לשפת פייטון חשוב היעילות מתכנת ויעילות מקום ולכן בחרה את עיקרון הזה על פני ה"נורמל אורדר"  
  
  
  
Nrmal order או במילים אחרות "לייזי אורדר" הוא עיקרון שנובע מהרצון לבדוק כמה שפחות למען מהירות הקוד הגבוה ביותר ולכן לא יוערכו כל הביטויים אלה רק מה שחייב אם לא חייב לא תתבצע בדיקה כגון חלקי הקוד בתנאים מקוננים בניגוד ל"אפלקטיב אורדר "   
שפת C העדיפה להשתמש במודל זה   
  
  
ד  
  
פונקציה רגילה היא הפשטה של תהליך זאת אומרת אני לא יודע מה מתבצע וזה לא מעניין אותי אני רק יודע מה אני אמור להכניס ומה אמור לצאת לי קופסא שחורה-מבחינה רעיונית ולכן אני לא מתחייב על ערכים  
  
  
פונקציה מסדר גבוהה היא או פונקציה המתקבלת כפרמט לפונקציה אחרת או פונקציה החוזרת כערך החזרה מפונקציה מופעלת ולכן אני לא מתחייב על הערכים מסוימים ובנוסף גם על הלוגיקה המתבצעת במימוש זה כאילו יש לי פעמיים קופסא שחורה ולכן רמת ההפשטה גדולה יותר