

מסמך זה מגדיר את קונבנציות הכתיבה של Verisoft על מנת להבטיח קוד קריא, תחזוקתי ועקבי בין כל המפתחים.

כללי בסיס

- כל שגיאה בקוד תיחשב ל-"error" ותימנע מקומפילציה עד לתיקונה.
- הקבצים שייבדקו הם קבצי `properties`, `java`, `-I`, `xml`.

1. אורך קוד ופורמט

- אורך הקובץ הכולל לא יעלה על 1000 שורות.
- אורך השורות בקוד Java לא יעלה על 120 תווים.
- אין להוסיף רווחים בסוף שורות קוד.
- על כל הקבצים להסתיים בשורת רווח חדשה.

2. הערות (Javadoc)

- כל שיטה ומחלקה חייבות לכלול הערות Javadoc לפי הסגנון והסטנדרטים המוגדרים:
 - הערות Javadoc נדרשות למיקום נכון בקוד (לפני ההצהרה על השיטה/מחלקה).
 - יש לשמור על סגנון אחיד של Javadoc לפי כללי הסגנון המקובלים (לדוגמה: תיאור קצר, תגי: `@return`, `param`, `@throws` במידת הצורך).
 - כל שיטה חייבת לכלול הערת Javadoc.

3. שמות משתנים ומתודות

- שמות המשתנים, המתודות, והקבועים יכתבו לפי קונבנציות כתיבה מוגדרות:
 - קבועים (Constants)** יכתבו באותיות גדולות עם קו תחתון (לדוגמה: `MAX_SIZE`).
 - שמות משתנים מקומיים** ופרמטרים יכתבו בפורמט `camelCase`.
 - שמות מחלקות** יתחילו באות גדולה ויוגדרו בשמות המתארים את מטרת המחלקה.

4. טיפול בייבוא (Imports)

- אין להשתמש בייבוא שאינו חוקי (למשל ייבוא של `*.sun`).
- יש להימנע מייבוא מיותר של ספריות או מחלקות שכבר לא נמצאות בשימוש בקוד.

5. מרווחים ושורות חדשות

- יש לשמור על רווחים נכונים:
 - אין להשאיר רווחים ריקים אחרי סימני פיסוק כמו סוגריים, נקודות פסיק או נקודות.
 - אין להשאיר רווחים לפני סימני פיסוק.
 - סוגריים מרובעים וסוגריים עגולים יכתבו צמודים לאלמנטים שהם עוטפים.

6. סדר modifiers

- יש להקפיד על סדר המחוונים (modifiers) בקוד, לדוגמה: `public`, `private`, `static`.
- אין להשתמש במודיפיירים מיותרים בקוד.

7. בלוקים של קוד (Blocks)

- יש להימנע מבלוקים מקוננים ללא צורך. קוד נקי וברור עדיף על קוד מסובך.
- אין להשאיר בלוקים ריקים – על כל בלוק תוכנית להכיל לפחות פקודה אחת.

- הסוגריים (} ו- {) יופיעו תמיד באותו קו עם ההצהרה או לפני התוכן, בהתאם למבנה הקוד.

8. בעיות קוד נפוצות

- יש להימנע מהשארת פקודות ריקות.
- חובה לוודא עקביות בין השיטות `equals` ו-`hashCode`. [להסבר מלא](#).
- יש להימנע משימוש לא נכון באובייקטים פנימיים או בהשמות בתוך ביטויים.
- במבני `switch`, חובה לכלול תנאי `default`.

9. עיצוב מחלקות

- הערות TODO יכתבו במקום המתאים תוך ציון תיאור ספציפי של מה נדרש להשלמה.
- יש להשתמש באות `L` גדולה לציון משתנה מסוג `long` כדי להימנע מבלבול עם הספרה 1.

10. Logging

- יש להימנע משימוש ב-`System.out.println` או `System.err.println`. במקום זאת, יש להשתמש ב-`logger` לצורך רישום מידע.
- יש לדווח בכל תחילת תהליך או עדכון חשוב.
- בתהליכים מורכבים עם רמת סיכון גבוהה לנפילה, יש להגדיל את תדירות הדיווח