

```

package anhang.version1;
import java.io.*;
import static anhang.Strings.*;

public class Wort implements Serializable {
    private final String wort;
    private static final long serialVersionUID = 1L;

    public Wort(String s) {
        if (!istWort(s)) throw new IllegalArgumentException();
        wort = s;
    }

    @Override
    public boolean equals(Object vergl) {
        if (vergl == this) return true;
        if (vergl == null) return false;
        if (vergl.getClass() != getClass()) return false;
        Wort w = (Wort) vergl;
        return w.wort.equals(wort);
    }

    @Override
    public int hashCode() {
        return wort.hashCode();
    }

    @Override
    public String toString() {
        return wort;
    }
}

```

---

```

package anhang.version2;

public class Wort implements Cloneable {
    protected String wort;

    public Wort(String s) {
        wort = s;
    }

    public void setString(String s) {
        wort = s;
    }

    @Override
    public boolean equals(Object vergl) {

```

```

        if (!(vergl instanceof Wort)) return false;
        Wort w = (Wort) vergl;
        return w.wort.equals(wort);
    }

    @Override
    public Wort clone() {
        try {
            return (Wort) super.clone();
        } catch (CloneNotSupportedException e) {
            throw new InternalError();
        }
    }

    @Override
    public String toString() {
        return wort;
    }
}

```

---

```

package anhang.version3;
import java.util.*;

public class Wort {
    final List<String> worte = new ArrayList<String>();

    public Wort(String... w) {
        for (String s : w) worte.add(s);
    }

    public void setString(String s) {
        worte.add(s);
    }

    public String toString() {
        return letzter();
    }

    public boolean equals(Wort vergl) {
        return vergl.letzter().equals(letzter());
    }

    private String letzter() {
        return worte.get(worte.size() - 1);
    }

    public void rueckgaengig() {
        worte.remove(worte.size() - 1);
    }
}

```

```
package anhang;
```

```
public final class Strings {
    private static final char BLANK = ' ';

    public static boolean istWort(String s) {
        if (s.isEmpty()) return false;
        return s.indexOf(BLANK) == -1;
    }
    public static boolean istHauptWort(String s) {
        if (!istWort(s)) return false;
        return Character.isUpperCase(s.charAt(0));
    }
}
```

```
package anhang;
import java.util.*;
```

```
public final class MusterStrings {
    public static final char BELIEBIGES_ZEICHEN = '?';
    private static final String BELIEBIGER_TEXT = "" + BELIEBIGES_ZEICHEN;

    public static boolean entsprichtMuster(String muster, String wort) {
        if (wort.length() != muster.length()) return false;
        for (int i = 0; i < muster.length(); i++) {
            char soll = muster.charAt(i);
            if (soll == BELIEBIGES_ZEICHEN) continue;
            if (wort.charAt(i) != soll) return false;
        }
        return true;
    }
    public static String wortNachMuster(String muster, char zeichen) {
        return muster.replace(BELIEBIGES_ZEICHEN, zeichen);
    }
    public static String wortNachMuster(String muster, String string) {
        return muster.replace(BELIEBIGER_TEXT, string);
    }
}
```

```
package anhang;
```

```
public enum Anrede {
    FRAU, HERR
}
```

```
package anhang;
import anhang.version?.Wort; // ? durch jeweilige Version ersetzen
```

```
public final class BleedsGschwaetz {
    private BleedsGschwaetz() {
    }

    public static boolean tuWasMit(Object o, Wort... worte) {
        if (o != worte[0]) return o.equals(worte[0]);
        return worte[0].equals(worte[1]);
    }
    public static Wort ehrenWort(final String s) {
        return new Wort(s) {
            @Override
            public boolean equals(Object vergl) {
                return vergl.toString().equalsIgnoreCase(s);
            }
        };
    }
    public static int version(Wort w) {
        final String v = "version";
        String name = w.getClass().toString();
        int index = name.indexOf(v);
        char c = name.charAt(index + v.length());
        int i = "0123".indexOf(c);
        return i;
    }
}
```