Code Smell 1:
Code Snippet:

```
//
// Source code recreated from a .class file by IntelliJ IDEA
// (powered by FernFlower decompiler)
//

package biz.ganttproject.core.calendar;

import biz.ganttproject.core.calendar.GPCalendarCalc.MoveDirection;
import biz.ganttproject.core.time.DateFrameable;
import biz.ganttproject.core.time.TimeDuration;
import biz.ganttproject.core.time.TimeUnit;
import com.google.common.collect.Lists;
import java.util.Date;
import java.util.Iterator;
import java.util.List;

abstract class GPCalendarBase implements GPCalendarCalc {
    private final List<GPCalendarListener> myListeners =
Lists.newArrayList();
    private String myName;
    private String myId;

    GPCalendarBase() {
    }
    (…)
    public Date findClosest(Date time, TimeUnit timeUnit,
GPCalendarCalc.MoveDirection direction, GPCalendar.DayType dayType, Date
limit) {
        return this.doFindClosest(time, timeUnit, direction, dayType,
limit);
    }
    protected Date doFindClosest(Date time, DateFrameable framer,
GPCalendarCalc.MoveDirection direction, GPCalendar.DayType dayType, Date
limit) {
        Date nextUnitStart = direction == MoveDirection.FORWARD ?
framer.adjustRight(time) : framer.jumpLeft(time);
        int nextUnitMask = this.getDayMask(nextUnitStart);
        switch (dayType) {
            case WORKING:
                if ((nextUnitMask & 1) == 1) {
                    return nextUnitStart;
                }
                break;
            case NON_WORKING:
            case WEEKEND:
            case HOLIDAY:
                if ((nextUnitMask & 1) == 0) {
                    return nextUnitStart;
                }
                break;
            default:
                assert false : "Should not be here";
        }
        return limit == null || (direction != MoveDirection.FORWARD ||
nextUnitStart.compareTo(limit) < 0) && (direction != MoveDirection.BACKWARD
|| nextUnitStart.compareTo(limit) > 0) ? this.doFindClosest(nextUnitStart,
framer, direction, dayType, limit) : null;
    }
//(…)
```

Location on the code base:
ganttproject\biz.ganttproject.core\bin\main\biz\ganttproject\core\calendar\GPCalendarBase.java

Code Smell identification: Speculative Generality – the method doFindClosest(…) has unnecessary code at the time in the switch (lines 93-96)

Refactoring proposal:
Eliminate the speculative code and add a relatable comment for the future when we implement the code

Code Smell 2
Code Snippet:

```
//(…)

    public void setPublicHolidays(Collection<CalendarEvent> holidays) {
    }

    public String getBaseCalendarID() {
        return null;
    }

    public void setBaseCalendarID(String id) {
    }

    public void importCalendar(GPCalendar calendar, ImportCalendarOption
importOption) {
    }
}
```

Location on the code base:
ganttproject\biz.ganttproject.core\bin\main\biz\ganttproject\core\calendar\AlwaysWorkingTimeCalendarImpl.java

Code Smell identification: Dead Code – Between lines 85 and 96, the methods are not finished and/or not used.

Refactoring proposal: Eliminate the dead code if it wouldn’t be used.

Code Smell 3:
Code Snippet:

```
//
// Source code recreated from a .class file by IntelliJ IDEA
// (powered by FernFlower decompiler)
//

package net.sourceforge.ganttproject;

public class PrjInfos {
    private String _sProjectName;
    private String _sDescription;
    private String _sOrganization;
    private String _sWebLink;

    public PrjInfos() {
        this._sProjectName = "Untitled Gantt Project";
        this._sDescription = "";
        this._sOrganization = "";
        this._sWebLink = "http://";
    }

    public PrjInfos(String sProjectName, String sDescription, String
sOrganization, String weblink) {
        this._sProjectName = sProjectName;
        this._sDescription = sDescription;
        this._sOrganization = sOrganization;
        this._sWebLink = sWebLink;
    }

    public String getName() {
        return this._sProjectName;
    }

    public void setName(String projectName) {
        this._sProjectName = projectName;
    }

    public String getDescription() {
        return this._sDescription;
    }

    public void setDescription(String description) {
        this._sDescription = description;
    }

    public String getOrganization() {
        return this._sOrganization;
    }

    public void set organization(String organization) {
        this._sOrganization = organization;
    }

    public String getWebLink() {
        return this._sWebLink;
    }

    public void setWebLink(String webLink) {
```

```
        this._sWebLink = webLink;
    }
}
```

Location on the code base:
ganttproject\bin\main\net\sourceforge\ganttproject\PrjInfos.class

Code Smell identification: Data Class – PrjInfos is a class only with getters and setters and with no real functionality.

Refactoring proposal: Review the code that uses the class. May find functionality that would be better located in the data class itself.