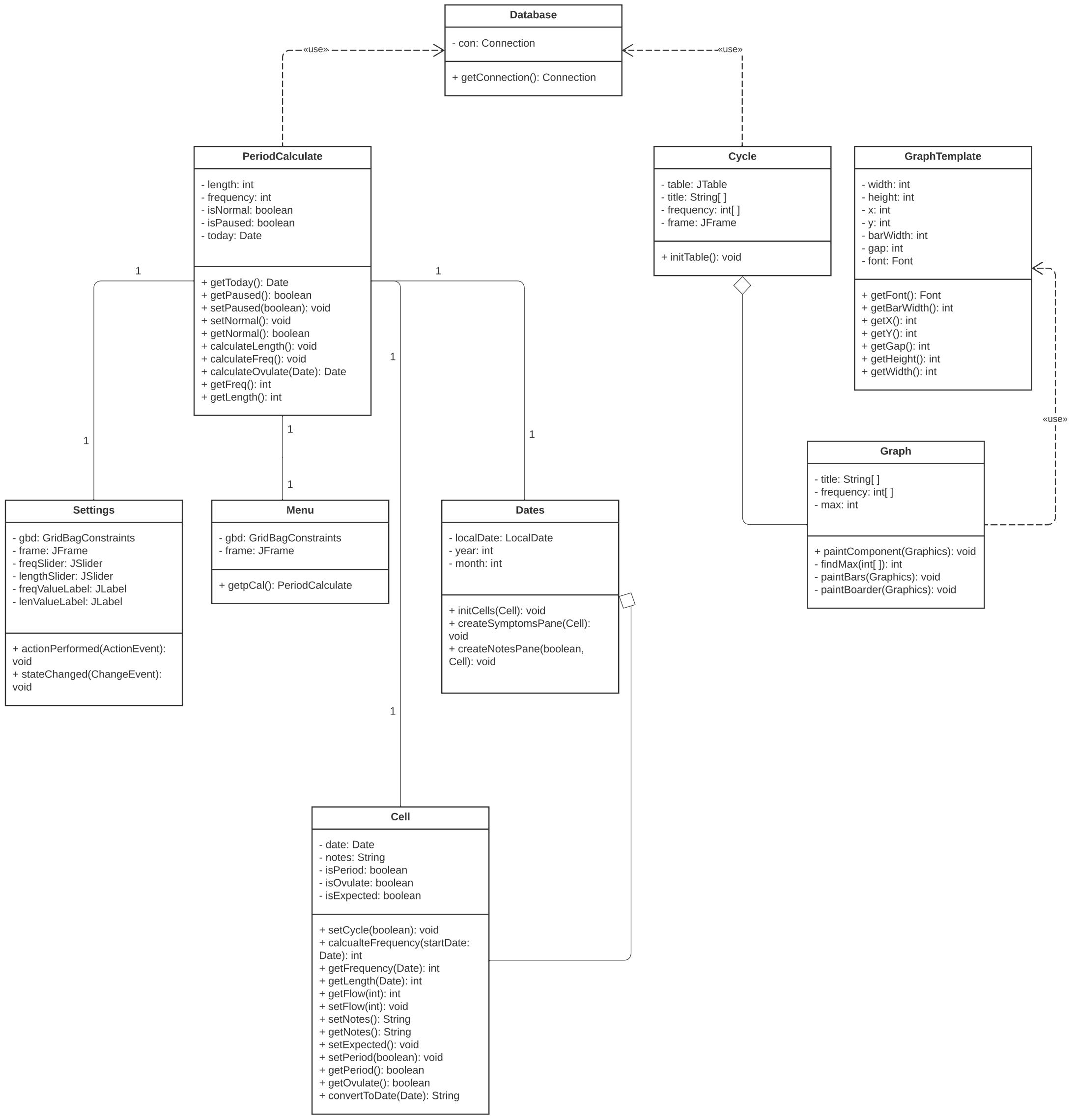
UML Diagram


Database
- con: Connection
+ getConnection(): Connection
«use»
«use»
PeriodCalculate
- length: int
- frequency: int
- isNormal: boolean
- isPaused: boolean
- today: Date
+ getToday(): Date
+ getPaused(): boolean
+ setPaused(boolean): void
+ setNormal(): void
+ getNormal(): boolean
+ calculateLength(): void
+ calculateFreq(): void
+ calculateOvulate(Date): Date
+ getFreq(): int
+ getLength(): int
Cycle
- table: JTable
- title: String[ ]
- frequency: int[ ]
- frame: JFrame
+ initTable(): void
GraphTemplate
- width: int
- height: int
- x: int
- y: int
- barWidth: int
- gap: int
- font: Font
+ getFont(): Font
+ getBarWidth(): int
+ getX(): int
+ getY(): int
+ getGap(): int
+ getHeight(): int
+ getWidth(): int
«use»
Graph
- title: String[ ]
- frequency: int[ ]
- max: int
+ paintComponent(Graphics): void
- findMax(int[ ]): int
- paintBars(Graphics): void
- paintBoarder(Graphics): void
Settings
- gbd: GridBagConstraints
- frame: JFrame
- freqSlider: JSlider
- lengthSlider: JSlider
- freqValueLabel: JLabel
- lenValueLabel: JLabel
+ actionPerformed(ActionEvent): void
+ stateChanged(ChangeEvent): void
Menu
- gbd: GridBagConstraints
- frame: JFrame
+ getpCal(): PeriodCalculate
Dates
- localDate: LocalDate
- year: int
- month: int
+ initCells(Cell): void
+ createSymptomsPane(Cell): void
+ createNotesPane(boolean, Cell): void
Cell
- date: Date
- notes: String
- isPeriod: boolean
- isOvulate: boolean
- isExpected: boolean
+ setCycle(boolean): void
+ calcualteFrequency(startDate: Date): int
+ getFrequency(Date): int
+ getLength(Date): int
+ getFlow(int): int
+ setFlow(int): void
+ setNotes(): String
+ getNotes(): String
+ setExpected(): void
+ setPeriod(boolean): void
+ getPeriod(): boolean
+ getOvulate(): boolean
+ convertToDate(Date): String
1
1
1
1
1
1
1
1
1