

研究ノート

JTreeを使用した医療用医薬品表示システムの試作

Prototype of JTree-based Drug Visualization System

大濱秀彦¹⁾, 豊田修一²⁾

Hidehiko Ohama, Shuichi Toyoda

要旨

政府の健康情報活用施策により、医療・健康情報を電子的に管理・活用できる情報サービスが実現されようとしている。一方、ITサービスにおいては、利用しやすいユーザインターフェイスの実現が重要になってくるため、情報視覚化技術が注目されている。そこで、本研究では、医療用医薬品の一覧を分かり易く表示するために、シュナイダーマンの情報視覚化のマントラやfocus+context技術を適用することを試みた。具体的には、類似薬選定の視点に着目したデータ構造を利用して、医療用医薬品データ視覚化方式を2種類提案した。これらを評価した結果、全体情報と詳細情報の表示の一貫性やバランスが十分とは言えないなどの課題を明らかにすることができた。

キーワード：医薬品表示システム、情報視覚化、Java

I. はじめに

政府の高度情報通信ネットワーク社会推進戦略本部は、平成22年に、医療情報化に関するタスクフォースを設置した。そして、医療・健康情報活用サービスとして、「どこでもMY病院」構想を検討している。この構想は、全国どこでも、個人が健康管理に取り組める環境を実現するため、国民が自らの医療・健康情報を電子的に管理・活用するための全国レベルの情報提供サービスを創出するものである(IT戦略本部、2010)。そして、健康に関心の深い個人が、医療情報を積極的に活用できるシステムの実現が望まれている。このような情報システムでは、専門知識を有さない一般人が利用することを前提とするため、使いやすいユーザインターフェイスの実現が求められる。

そこで、薬剤情報の利用を促進することを目的として、医療用医薬品を類似薬選定の視点から表示するシステムの開発をはじめた。本稿では、インター

ネットの普及に伴い急速に普及したJava言語をベースにした医療用医薬品の一覧表示システムの研究開発について述べる。II章では本研究の関連技術について、III章ではJavaについて、IV章ではシステム開発について述べる。

II. 関連技術

ここでは、関連技術として、情報視覚化とオブジェクト指向について記述する。

1. 情報視覚化

今後のITサービスは、機能を提供するだけのユーザインターフェイスから利用しやすさを考慮したユーザインターフェイスへ変化していくと考えられる。情報視覚化(Information Visualization)は、多くの情報を、ユーザにわかりやすく提示するための技術の一つであり、その重要性が高まりつつある。情報視覚化は、検索の省力化やパターン検出の容易

1) アカギシステムズ株式会社、2) 上武大学看護学部

化を実現する技術であり、さまざまな手法が提案されている。代表的なものには、表示空間の拡張、レイアウト最適化、空間充てん型表示、フォーカス+コンテキスト (focus+context) 技術などがある (Spence, 2007)。

表示空間の拡張では、より多くの情報を表示するために、通常のコンピュータ画面よりも大きな表示空間を用意する解決法がある。複数の仮想画面を切り替えたり、奥行きを利用するための3次元表示をしたり、半透明の層を重ねて情報を表示したりする方法である。レイアウト最適化は、図形を線でつないだグラフ構造を自動的に描画する場合などに使用される。ノードやリンクの交差が少なくなるように最適化する。

空間充てん型表示は、表示空間を分割するようにして、むだなスペースの割合を最小限にして、効率よく情報を詰め込んで表示するものである。これには、情報を表すオブジェクトの形状を規格化して、タイル張りのように敷き詰める方法などがある。focus+context技術は、注目点の付近の局所的詳細を表示しながら、周囲の大局的外観も同時に表示する技術である (岡田ら、2002)。

情報視覚化の特徴を端的に表しているのが、シュナイダーマン (Shneiderman) のマントラ “Overview first, zoom and filter, then details on demand.” である。情報をわかりやすく表示するためには、ユーザにまず情報全体の概略を示し、ユーザ要求に応じてその中から重要な情報を拡大・選別するなどして、さらに詳細な情報を提供する、ということを主張している (Card et al, 1999)。

2. オブジェクト指向

オブジェクト指向とは、実世界に存在する物や概念をオブジェクトとして捉える考え方である。ソフトウェアの設計や開発においては、操作の手順よりもオブジェクト間の相互作用をソフトウェアの動作と捉える。プログラミング言語で考えるオブジェクトとは、日常に見られる物体を、ソフトウェアの中でモデル化して表現したものである。このオブジェクトを用いて、物や概念をプログラム上に表現する。

自動車をオブジェクト指向の考え方で捉えると、ひとつひとつの部品をオブジェクトと考えることができる。そして、独立した部品 (オブジェクト) が集まって自動車を形成している。各部品はそれぞれ

固有の役割をもち、部品間で連携しながら全体として自動車を動作させる。このような形態がオブジェクト指向の基本的な考え方になる。

クラスは、オブジェクトを作るための設計図である。同じ特性をもつオブジェクトに名前を付けて、データ (フィールド) と関数 (メソッド) の集合を一つの型として宣言したものである。また、クラスは、オブジェクトの特性を抽象化したものともいえる。クラスはオブジェクトを生成するために宣言した型であり、オブジェクトの性質や動作をまとめたものである。そして、オブジェクトは、クラスから作られる (布広、2008)。

III. Java

ここでは、Javaの概要、GUI (Graphical User Interface) 機能、階層構造のデータを取り扱えるJTree、データベース、イベント処理について記述する。

1. 概要

Javaは、インターネットの普及にともない、急速に広まったプログラミング言語である。Javaは以下のような特徴を有する。

- ・プラットフォームを選ばない
- ・オブジェクト指向言語である
- ・プログラムを安全に実行できる
- ・ネットワーク関連の機能が充実している

Javaで記述されたプログラムは、Java仮想マシン (Java Virtual Machine) 上で動作する。Javaのソースコードをコンパイルすると、バイトコードと呼ばれる中間コードに変換される。バイトコードが書かれたファイルをクラスファイルという。このクラスファイルをJava仮想マシンが各プラットフォームに対応したネイティブコードに変換して実行する。つまり、Java仮想マシンがプラットフォーム間の違いを吸収する役割を担っている。このため、Javaで作られたプログラムは、特定のOSやCPUに依存することなく配布でき、“Write once, run anywhere”といわれる。

2. GUI

Javaには、Swingと、そのベースになったAWT (Abstract Window Toolkit) の2つのGUIライブラリが存在する。AWTは、Javaの当初から用意されている標準的ライブラリである。GUI部品はプラット

フォーム依存のネイティブ部品として実装されている。Swingは、Java2から導入された。プラットフォームに依存のないコードで実現されている。Javaでは、GUI部品は総称してコンポーネントと言う。

Swingのコンポーネントは、3つの特性を持っている。ボタン状態やテキストフィールドに入力されたテキストなどのコンテンツ、色やサイズなどの外観、イベントに対応する動作である。

また、Swingは、MVC設計パターンを採用している。MVC設計パターンは、1つのオブジェクトに過剰な役割を与えないという考え方である。この設計パターンでは、モデル (Model)、ビュー (View)、コントローラ (Controller) 3つの異なるクラスが実装される。モデルは、コンテンツを格納するものであり、ユーザインターフェイスに関するパターンは全く持たない。ビューは、モデルに格納されたコンテンツ (データ) を画面に描く。コントローラは、マウスクリックやキー入力などのユーザ入力イベントを処理する。

Swingのコンポーネントは、最上位コンテナ、中間コンテナ、アトミック・コンポーネントに分類できる。最上位コンテナは、GUI部品を画面上に表示するために必要なコンポーネントである。最上位コンテナには、GUI部品を画面上に表示する領域であるコンテンツ・ペインと呼ばれる領域がある。中間コンテナは、他のコンポーネント配置などのためのコンポーネントである。アトミック・コンポーネントは、個々のGUI部品のことであり、ユーザ入力を受け付けるコンポーネントである基本コントロール、ユーザに情報を提示するコンポーネントである編集不能情報ディスプレイ、さまざまな編集機能と対話性をもつコンポーネントである対話型高度情報ディスプレイがある。代表的なコンポーネントを表1に示す。

Swingでは、この3つの階層のコンポーネントを組み合わせてアプリケーションを作成する。例えば、JLabelなどのコンポーネントを持ったアプリケーションを作成するためには、まず最上位レベルコンテナを作成し、そこに中間コンテナを配置し、さらに中間コンテナにコンポーネントを配置するといった手順を踏む必要がある (赤間、2008)。

3. JTree

JTreeは、構造を持ったデータを扱うSwingのコンポーネントの一つである。JListやJTableも構造を

表1. Swingの代表的コンポーネント

種 類	コンポーネント	目 的
最上位 コンテナ	JFrame	メインウインドウ作成
	JDialog	ダイアログ作成
	JApplet	アプレット作成
中間 コンテナ	JPanel	コンポーネント配置
	JScrollPane	スクロールバー表示
アトミック コンテナ	JButton	ボタン
	JCheckBox	項目選択
	JSlider	数値指定
	JLabel	文字表示領域

持ったデータを扱うSwingのコンポーネントである。JTreeは階層構造を持ったデータを扱うためのものである。JTreeでは、表示される一つ一つのデータはノードと呼ばれる。一番土台となる部分 (ルート) のノードが1つあり、この下にノードを追加していくことで、データ全体を構築する。標準的に使用するDefaultMutableTreeModel クラスは、子ノードを追加・挿入・削除するなどの機能を有している。さらに、下位のノードを展開する機能や折りたたむ機能、指定されたノードのパスも出力する機能なども有する。また、ノードには、テキストだけでなく、アイコンも設定することができる。このような機能により、JTreeを利用することで、木構造をダイナミックに構成したり、木構造を利用してデータをアクセスすることが実現できる (大村、2008)。

4. データベースとイベント処理

Javaでは関係データベースを利用する時の標準APIとして、JDBC (Java DataBase Connectivity) が準備されている。JDBCを利用すると、特定のデータベースに依存しないプログラムが開発できる。JDBCのAPIでは、関係データベースに接続、レコード検索、レコード更新ができる (井上ら、2009)。

また、GUI環境では、キーボードやマウスからのユーザ操作をイベントと呼ぶ。プログラムはイベントに応じた処理を準備する必要がある。Javaのイベント処理は、イベントソース、イベントリスナ、イベントハンドラの3者の相互作用で実現する。イベントソースは、イベントの源になるものである。マウスクリックやキーボード入力を受け、イベントを

生成するコンポーネントのことである。イベントリスナは、発生したイベントを受け取り、イベントに反応できる機能を有するものである。イベントソースにイベントリスナを登録することで、両者は接続される。イベントハンドラは、イベントソースから送られてきたイベントを実際に処理するものである（溝口ら、1999）。

IV. システム開発

1. 概要

医師からの処方箋などによって使用される医療用医薬品は、一般名で2600件を、商品数で17000件を超えている。このような医療用医薬品は、図1のように、類似薬選定の視点に着目して、薬効別薬剤分類層・一般名分類層・商品名層の3層構造で分類することが多い。ここでは、この3層構造を意味的分類構造と呼ぶことにする。意味的分類構造には、深さが浅く、最下位層の幅が極めて広いという特徴がある。このため、分類されたデータを一覧表示しようとすると、コンピュータ画面の物理的制約から単純な手法で実現できるものではない。そこで、本研究では、意味的分類構造に着目して医療用医薬品の一覧を分かり易く表示するために、シュナイダーマンのマントラとfocus+context技術の適用を試みた。

2. 開発環境

本システムは、次のような開発環境で実施した。プログラミング言語としてはオブジェクト指向言語であるJava SE6を採用した。GUIのクラスライブラ

リとしてはSwingを使用し、木構造の表示コンポーネントとしてはJTreeを利用した。統合開発環境にはEclipse 3.4を用いた。薬剤データを管理するデータベース管理システム（DBMS）にはMySQL 5.2を採用し、データベースアクセスにはJavaの標準APIであるJDBCを使用した。なお、本システムは、ローカルコンピュータ上で動作するJavaアプリケーションプログラムである（掌田、2008）。

3. データ構造

本研究で開発したシステムは、医薬品データベースを用いて、医薬品の意味的分類構造に対応した分類データ木を構築し、医薬品データの表示を行っている。

医薬品の分類データ木は、第1階層のノードに薬効分類を配置し、各薬効分類に対応する一般名を第2階層のノードに配置している。さらに、各一般名に対応する商品名を第3階層のノードに配置している。このような木構造を構築することで、同一階層の兄弟ノード、前後の階層に対応する親ノードや子ノードを辿ることができるようになる。図2は、医薬品の分類データ木の構築例である。

4. 表示方式

本システムでは、単一木表示方式と複数木表示方式の2種類の医療用医薬品の一覧表示方式を提案する。単一木表示方式は、薬効別薬剤分類、一般名、商品名の全てのデータを単一木の形式で表示する方式である。薬効別薬剤分類層、一般名分類層、商品名層3層を順次展開できる機能を実現した。図3は、薬効別薬剤分類層、一般名分類層、商品名層と順次展開していく様子を示す。つまり、シュナイダーマンのマントラが主張するように、全体情報から詳細情報へ展開している。

複数木表示方式は、データを薬効分類木と薬剤項目木の2つの木に分割して表示する方式である。薬剤分類木では大局的外観を表示し、薬剤項目木では局所的詳細を表示する。つまり、focus+context技術を適用したものである。薬剤分類木は、薬効別薬剤分類と一般名のデータを表示対象としている。薬剤項目木は、薬剤分類木で選択された一般名に対応した商品名データを表示対象としている。また、薬剤分類木で選択された一般名から木構造を辿って前後の一般名を確定することができる。そこで、薬剤項

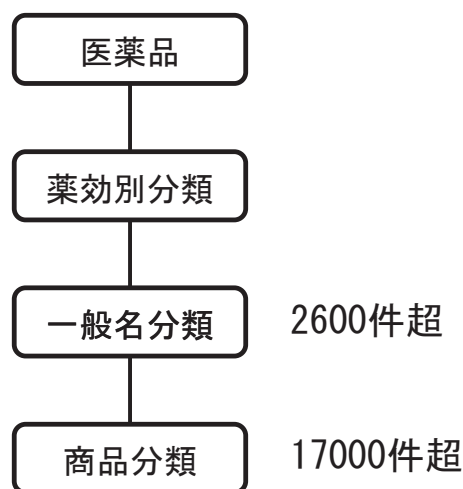


図1. 医薬品分類のための3層構造

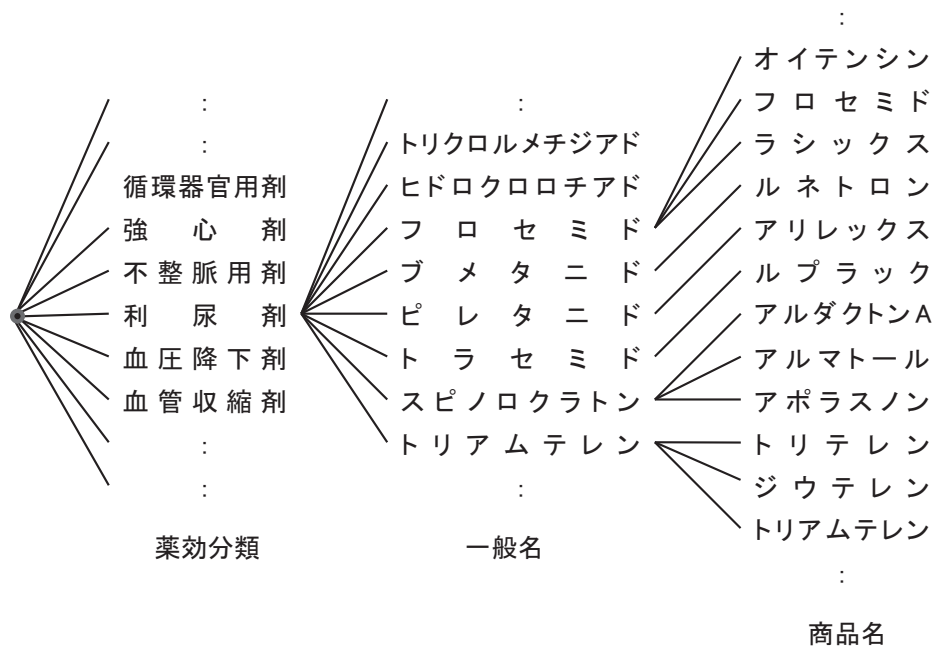


図 2. 医薬品の分類データ木の構築例

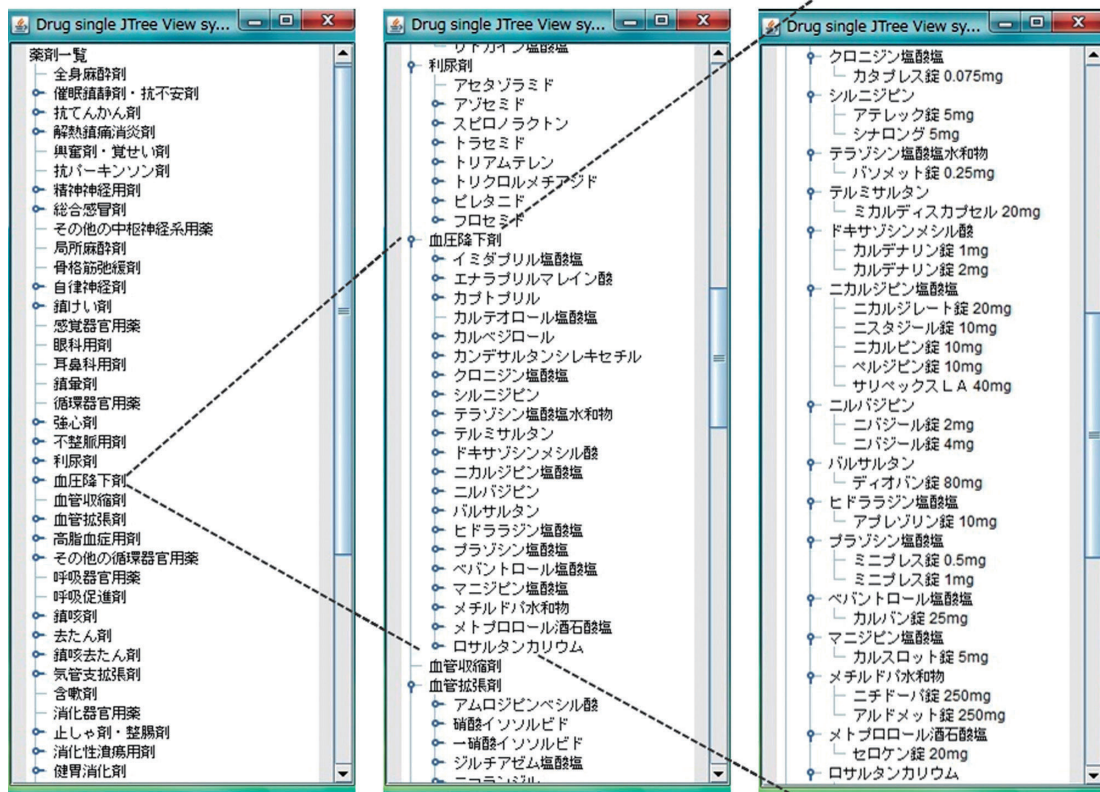


図 3. 単一木表示方式のサンプル画面



図 4. 複数木表示方式のサンプル画面（薬効分類のみ表示）

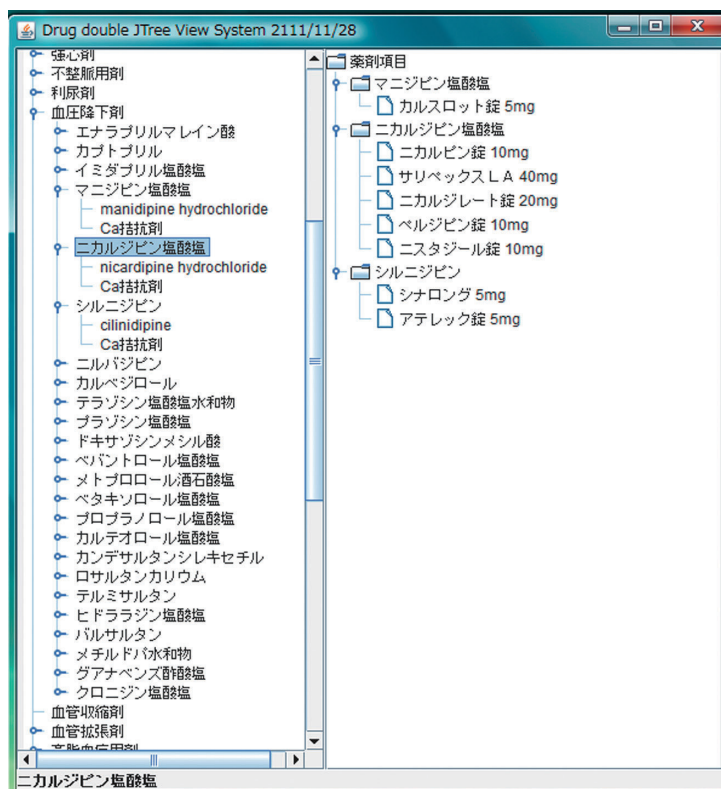


図 5. 複数木表示方式のサンプル画面（全体表示）

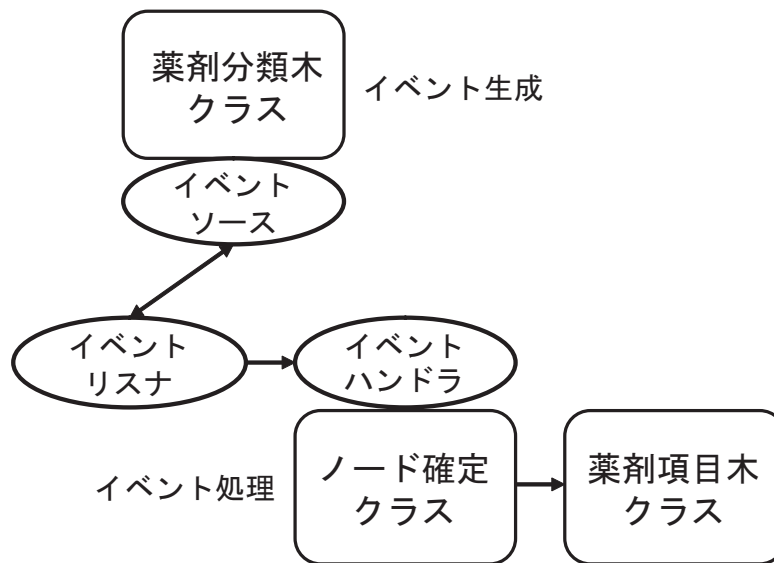


図6. イベント処理の概要

目木では、前後の一般名に対応する商品名データも表示している。図4は複数木表示方式で薬効別薬剤分類だけを展開したサンプル画面である。図5は薬剤分類木と薬剤項目木を同時に表示しているサンプル画面である。この画面では、薬剤分類木においてカリシウム拮抗剤系の血圧降下剤であるニカルジピン塩酸塩を選択し、薬剤項目木では関連する商品名を表示している。なお、薬効分類木と薬剤項目木の同期は、イベント処理により実現している。このイベント処理の概要を図6に示す。これにより、薬効分類木で選択された一般名の情報が薬剤項目木に伝えられ、連携した動作を実現している。

5. 評価

単一木表示方式では、最下位層である商品名層まで展開すると、薬効別薬剤分類に関する周辺情報が画面の物理的制約から一度に表示できない可能性が高い。これらの情報を確認するためには、スクロール操作を多用する必要性が生じる。一方、複数木表示方式は、商品名層の表示情報をコントロールする方式である。単一木表示方式と比較すると、画面の物理的制約を補う方式である。しかしながら、本システムで実現した方式では、全体情報と詳細情報の表示の一貫性やバランスが十分とは言えず、今後の

課題である。

V. おわりに

インターネットと相性の良い開発環境には、PHPやJavaがある。本システムの研究開発では、情報の視覚化に重点があったため、GUI機能が豊富なJavaを利用した。しかしながら、本システムでは、豊富なGUI機能の一部分しか利用していない。今後は、多くのGUI機能を利用しながら、本研究で明らかになった課題を解決していきたい。

参考文献

1. IT戦略本部 (2010) : 高度情報通信ネットワーク社会推進戦略本部 (IT戦略本部) 医療情報化に関するタスクフォース (官邸)
2. 岡田謙一、西田正吾、葛岡英明、仲谷美江、塩澤秀和 (2002) : ヒューマンコンピュータインタラクション、オーム社
3. Stuart K. Card, Jock D. Mackinlay, Ben Shneiderman (1999) : Readings in Information Visualization Using Vision to Think, Morgan Kaufmann Publishers.
4. Robert Spence (2007) : Representation, Information Visualization Design for Interaction (Second

- Edition), Pearson Education
5. 布広永示 (2008) : Javaオブジェクト指向プログラミング、オーム社
 6. 赤間世紀 (2008) : Java Swing 教科書、工学社
 7. 大村忠史、池田成樹 (2008) : Java GUI プログラミング Java SE6 対応 Vol.II、カットシステム
 8. 井上誠一郎、永井雅人、松山智大 (2009) : パーフェクトJava、技術評論社
 9. 溝口文雄、大和田勇人 (1999) : 入れたてJava (第2版)、共立出版
 10. 掌田津耶乃 (2008) : Eclipse 3ではじめるJavaプログラミング入門、秀和システム