

れ以外の用法区分の場合は 0 を返す。

### 1.10.7. 投薬経路

#### ■定義

投薬経路を表現する投薬経路コード。

ValueSet: Kobe.Pharmacy.ValueSets.RouteOfAdministration  
を使用する。CE 型。

#### ■現行対応項目:

- 対応なし。

#### ■インターフェース

**object IMedicationAdministration.RouteCode [get;]**

投薬経路コードを object として取得する。

ValueSet: Kobe.Pharmacy.ValueSets.RouteOfAdministration で定義されたコード値で、実際の型は CE 型である。

コードの文字列表現を得たい場合には、

```
string RouteOfAdministration.Singleton.ToString(object);
```

を使用する。

コードの表示用文字列を得たい場合には、

```
string RouteOfAdministration.Singleton.DisplayNameOf(object);
```

を使用する。

特定のコードとの等価性を調べる場合には、

```
bool RouteOfAdministration.Singleton.<Get property for a specific code>.Equals(object);
```

を使用する。

**string IMedicationAdministration.RouteCodeText [get;]**

投薬経路コードのオリジナルテキストを取得する。値がない場合は null を返す。

検討。必要か？

### 1.10.8. 用法区分コード

#### ■定義

用法の種別を表現する用法区分コード。

ValueSet: Kobe.Pharmacy.ValueSets.PrescriptionProductUsageType  
を使用する。CD 型。

■現行対応項目:

- 用法区分 (ORDERYOH\_6)

■インターフェース

**object IMedicationAdministration.UsageType [get;]**

用法区分コードを object として取得する。

ValueSet: Kobe.Pharmacy.ValueSets.PrescriptionProductUsageType で定義され  
たコード値で、実際の型は CD 型である。

コードの文字列表現を得たい場合には、

```
string ClinicalDrugUsageType.Singleton.ToString(object);
```

を使用する。

コードの表示用文字列を得たい場合には、

```
string ClinicalDrugUsageType.Singleton.DisplayNameOf(object);
```

を使用する。

特定のコードとの等価性を調べる場合には、

```
bool ClinicalDrugUsageType.Singleton.<Get property for a specific  
code>.Equals(object);
```

を使用する。

### 1.10.9. 用法コード

■定義

用法コード。最大 2 つまで指定可能。

ValueSet: Kobe.Pharmacy.ValueSets.PrescriptionProductUsageType  
を使用する。CD 型。

■現行対応項目:

- コード 1 (ORDERYOH\_7)
- コード 2 (ORDERYOH\_8)

■インターフェース

**object[] IMedicationAdministration.ArrayOfUsageCode [get;]**

用法区分コードを object の配列として取得する。

ValueSet: `Kobe.Pharmacy.ValueSets.PrescriptionProductUsageType` で定義されたコード値で、実際の型は CD 型である。

コードの文字列表現を得たい場合には、

```
string ClinicalDrugUsageCode.Singleton.ToString(object);
```

を使用する。

コードの表示用文字列を得たい場合には、

```
string ClinicalDrugUsageCode.Singleton.DisplayNameOf(object);
```

を使用する。

特定のコードとの等価性を調べる場合には、

```
bool ClinicalDrugUsageCode.Singleton.<Get property for a specific code>.Equals(object);
```

を使用する。

#### 1.10.10. 用法名称

##### ■定義

指定された用法コードから作成される用法名称。

##### ■現行対応項目:

- 用法名称 (ORDERYOH\_10)

##### ■インターフェース

```
string IMedicationAdministration.UsageName [get;]
```

用法名称を取得する。値がない場合には null を返す。

#### 1.10.11. 服用開始区分

##### ■定義

入院処方オーダーにおいて、服用開始時間帯（朝、昼、夜）を指定するためのコード。

ValueSet: `Kobe.Pharmacy.ValueSets.AdministrationStartTimingCode` を使用する。CD 型。入院メッセージのみ。外来メッセージには使用しない。

##### ■現行対応項目:

- 服用開始区分 (ORDERCTL\_36)

## ■インターフェース

**object IMedicationAdministration.StartTimingCode [get;]**

服用開始区分コードを object として取得する。

ValueSet: Kobe.Pharmacy.ValueSets.AdministrationStartTimingCode で定義されたコード値で、実際の型は CD 型である。

コードの文字列表現を得たい場合には、

```
string AdministrationStartTimingCode.Singleton.ToString(object);
```

を使用する。

コードの表示用文字列を得たい場合には、

```
string
```

```
AdministrationStartTimingCode.Singleton.DisplayNameOf(object);
```

を使用する。

特定のコードとの等価性を調べる場合には、

```
bool AdministrationStartTimingCode.Singleton.<Get property for a specific code>.Equals(object);
```

を使用する。

### 1.10.12. 隔日指示コード

#### ■定義

服用間隔を表す隔日指示を指定するためのコード。

ValueSet: Kobe.Pharmacy.ValueSets.AdministrationRepeatPattern  
を使用する。CD 型。特に指示がない場合は、「毎日」を示すコードを指定する。

#### ■現行対応項目:

- 隔日指示 (ORDERYOH\_29)

#### ■インターフェース

**object IMedicationAdministration.RepeatPatternCode [get;]**

隔日指示コードを object として取得する。

ValueSet: Kobe.Pharmacy.ValueSets.AdministrationRepeatPattern で定義されたコード値で、実際の型は CD 型である。

コードの文字列表現を得たい場合には、

```
string AdministrationRepeatPattern.Singleton.ToString(object);
```

を使用する。

コードの表示用文字列を得たい場合には、

string

```
AdministrationRepeatPattern.Singleton.DisplayNameOf(object);
```

を使用する。

特定のコードとの等価性を調べる場合には、

```
bool AdministrationRepeatPattern.Singleton.<Get property for a  
specific code>.Equals(object);
```

を使用する。

### 1.10.13. 用法指示コメント

#### ■定義

処方箋内のある特定の用法に対する患者向けコメント。

#### ■現行対応項目:

- 指示種別 (ORDERSIJ\_6) (患者)
- 指示コメント (ORDERSIJ\_8)

#### ■インターフェース

```
string IMedicationAdministration.Instruction [get;]
```

特定の用法のための患者向け指示コメントを取得する。値がない場合には null を返す。

### 1.10.14. 服用指示コメント (指導コメント)

#### ■定義

ユヤマが薬袋に投薬・服用指示を印字する際に使用するコード。

#### ■現行対応項目:

- 新規追加

#### ■インターフェース

```
object[] IMedicationAdministration.ArrayOfInstructionCode [get;]
```

服用指示コメント (指導コメント) を取得する。値がない場合には null を返す。

### 1.10.15. 警告情報

#### ■定義

警告に関する情報。詳細は、〇〇〇節の警告情報参照。

#### ■現行対応項目:

- なし

#### ■インターフェース

**int IMedicationAdministration.AlertCount [get;]**

警報数を取得する。

**IAlert IMedicationAdministration.AlertAt(int i)**

警告情報を、IAlert インターフェースとして取得する。対応する警告情報が存在しない場合には null を返す。

## 1.11. 警告情報

警告に関する情報。

### 1.11.1. 警告コード

#### ■定義

禁忌などのオーダー時の警告を表すコード。HL7 で決められたコードを使用する警告分類コードと、ローカルで定義されたコードを使用可能な警告詳細コードが在る。

警告分類コードは、

```
ValueSet: Nori.HL7.VocabularyDomains.ActCode  
(ActSuppliedItemDetectedIssueCode)
```

を使用する。CS 型。

警告詳細コードは、

```
ValueSet: Kobe.Pharmacy.ValueSets.AdministrationDetectedIssueDetailCode
```

を使用する。CE 型。

#### ■現行対応項目:

- 警告コード (ORDERYAK\_22)

#### ■インターフェース

```
object IAlert.Code [get;]
```

警告分類コードを object として取得する。

ValueSet: Nori.HL7.VocabularyDomains.ActCode で定義されたコード値で、実際の型は CS 型である。

コードの文字列表現を得たい場合には、

```
string ActCode.Singleton.ToString(object);
```

を使用する。

コードの表示用文字列を得たい場合には、

```
string ActCode.Singleton.DisplayNameOf(object);
```

を使用する。

特定のコードとの等価性を調べる場合には、

```
bool ActCode.Singleton.<Get property for a specific  
code>.Equals(object);
```

を使用する。

**object IAlert. DetailCode [get;]**

警告詳細コードを object として取得する。

ValueSet: Kobe.Pharmacy.ValueSets.AdministrationDetectedIssueDetailCode で定義されたコード値で、実際の型は CE 型である。

コードの文字列表現を得たい場合には、

**string**

**AdministrationDetectedIssueDetailCode.Singleton.ToString(object);**

を使用する。

コードの表示用文字列を得たい場合には、

**string**

**AdministrationDetectedIssueDetailCode.Singleton.DisplayNameOf(object);**

を使用する。

特定のコードとの等価性を調べる場合には、

**bool AdministrationDetectedIssueDetailCode.Singleton.<Get property for a specific code>.Equals(object);**

を使用する。



## 1.12. 健康保険情報

### 1.12.1. 保険 ID

#### ■定義

健康保険情報を一意に識別するコード。

ValueSet: Kobe.Pharmacy.ValueSets.InsuranceCode  
を使用する。CD 型。

保険 ID と法制コードは、ひとつのコード体系で表現されることに注意。

#### ■現行対応項目:

- 保険 ID (INSURANCE\_4)
- 法制区分 (INSURANCE\_5)

#### ■インターフェース

**object IInsurance.Code [get;]**

保険 ID (コード) を object として取得する。

ValueSet: Kobe.Pharmacy.ValueSets.InsuranceCode で定義されたコード値で、実際の型は CD 型である。

コードの文字列表現を得たい場合には、

```
string InsuranceCode.Singleton.ToString(object);
```

を使用する。

コードの表示用文字列を得たい場合には、

```
string InsuranceCode.Singleton.DisplayNameOf(object);
```

を使用する。

特定のコードとの等価性を調べる場合には、

```
bool InsuranceCode.Singleton.<Get property for a specific  
code>.Equals(object);
```

を使用する。

#### ■HL7 マッピング(参考)

OrderSetHeader.pertainsTo.PolicyOrAccount の code 属性。

XPath: OrderSetHeader/pertainsTo/policyOrAccount/code

### 1.12.2. 有効期限

#### ■定義

健康保険の有効期限。

#### ■現行対応項目:

- 有効期限（新規追加）

#### ■インターフェース

**DateTime IInsurance.IssuedDate [get;]**

健康保険の有効期限の開始年月日日(発行日)を返す。ない場合は new DateTime()を返す。

**DateTime IInsurance.ExpirationDate [get;]**

健康保険の有効期限の終了年月日を返す。ない場合は new DateTime()を返す。

### 1.12.3. 被保険者記号・番号

#### ■定義

健康保険の被保険者記号・番号。

#### ■現行対応項目:

- 被保険者記号（新規追加）
- 被保険者番号（新規追加）

#### ■インターフェース

**string IInsurance.InsuredPersonSymbol [get;]** (HL7 必須属性)

健康保険の被保険者記号を取得する。ない場合は null を返す。

**string IInsurance.InsuredPersonNumber [get;]** (HL7 必須属性)

健康保険の被保険者番号を取得する。ない場合は null を返す。

**string IInsurance.InsuredPersonRootUId [get;]** (HL7 必須属性)

健康保険の被保険者記号・番号の(RootUID)を取得する。RootUId を持たない場合には null を返す。

**string IInsurance.InsuredPersonIdIssuingAuthorityName [get;]**

健康保険の被保険者記号・番号の ID 発行機関名称を取得する。ID 発行機関名称を持たない場合には null を返す。

■HL7 マッピング(参考)

OrderSetHeader.pertainsTo.PolicyOrAccount.beneficiary.CoveredParty の id 属性。

XPath:

```
OrderSetHeader/pertainsTo/policyOrAccount/beneficiary/coveredParty  
/id[0]/@extension
```

InsuredPersonSymbol と InsuredPersonNumber は、1 つの II 型値としてマッピングされるため、InsuredPersonSymbol と InsuredPersonNumber をアンダスコア()で合成した文字列を II 型の extension とする。メッセージタイプから取り出す場合は、アンダスコアで split し、それぞれの値とする。

coveredParty.code には、RoleCode の CoverageRoleType を入れることになっており、保険の適用が本人か家族かどうかを区別するコードを入れることになっている。今回の場合、CE のヌルインスタンス(NoInformation)を指定する。

coverageParty.name には、IPatient インターフェースからとってきた患者名を入れる。

coverageParty.multipleBirthInd には、ヌルインスタンス(No Information)を入れる。

#### 1.12.4. 保険者コード

■定義

健康保険の保険者コード。

■現行対応項目:

- 保険者コード (新規追加)

■インターフェース

**string IInsurance.ProviderId [get;]** (HL7 必須属性)

健康保険の保険者コードを取得する。ない場合は null を返す。

**string IInsurance.ProviderRootUid [get;]** (HL7 必須属性)

健康保険の保険者コードの RootUid を取得する。RootUid を持たない場合には null を返

す。

**string IInsurance.InsuredPersonIdIssuingAuthorityName [get;]**

健康保険の保険者コードの ID 発行機関名称を取得する。ID 発行機関名称を持たない場合には null を返す。

■HL7 マッピング(参考)

OrderSetHeader.pertainsTo.PolicyOrAccount.author.CarrierRole の id 属性。

XPath:

```
OrderSetHeader/pertainsTo/policyOrAccount/author/carrierRole/id[0]
/@extension
```

### 1.12.5. 給付割合

■定義

健康保険の保険会社の外来時負担率、入院時負担率。0~1 の値で定義。

■現行対応項目:

- 給付割合 (外来) (新規追加)

■インターフェース

**double IInsurance.RateOnAdmission [get;]**

健康保険の入院時負担率を取得する。値がない場合は new double() を返す。明示的に Null インスタンスを設定したい場合は、double.NaN を指定する。

**double IInsurance.RateOnOutpatient [get;]**

健康保険の外来時負担率を取得する。値がない場合は new double() を返す。明示的に Null インスタンスを設定したい場合は、double.NaN を指定する。

■HL7 マッピング(参考)

OrderSetHeader.pertainsTo.PolicyOrAccount.coverageOf.invoiceElementDetail の factorNumber 属性。

XPath:

```
OrderSetHeader/pertainsTo/policyOrAccount/coverageOf/invoiceElementDetail/factorNumber/@value
```

入院時負担率: invoiceElementDetail.code ==  
    InvoiceElementDetailType.RateOnAdmission  
外来時負担率 invoiceElementDetail.code ==  
    InvoiceElementDetailType.RateOnOutpatient

## 1.13. 病名情報

ユヤマ向けメッセージのための病名情報。

### 1.13.1. 病名識別情報

#### ■定義

病名オーダーを識別する ID。

#### ■現行対応項目:

- なし

#### ■インターフェース

**string IDiagnosis.Id [get;]** (HL7 必須属性)

病名オーダーID(Extension)を取得する。病名オーダーID (Extension)を持たない場合には null を返す。

**string IDiagnosis.RootUId [get;]** (HL7 必須属性)

病名オーダーID (RootUID)を取得する。病名オーダーID (RootUID)を持たない場合には null を返す。

**string IDiagnosis.IdIssuingAuthorityName [get;]**

病名オーダーID 発行機関名称を取得する。ID 発行機関名称を持たない場合には null を返す。

#### ■HL7 マッピング(参考)

OrderSetHeader.pertinentInformation.pertinentObservation の id 属性。

XPath:

```
OrderSetHeader/pertinentInformation/pertinentObservation/id[0]/@extension
```

### 1.13.2. 病名区分情報

#### ■定義

病名区分を表すコード。

ValueSet: Kobe.Pharmacy.ValueSets.ObservationDiagnosisType

を使用する。CE 型。

■現行対応項目:

- なし

■インターフェース

**object IDiagnosis.CategoryCode [get;]** (HL7 必須属性)

病名区分コードを object として取得する。

ValueSet: Kobe.Pharmacy.ValueSets. ObservationDiagnosisType で定義されたコード値で、実際の型は CE 型である。

■HL7 マッピング(参考)

OrderSetHeader.pertinentInformation.pertinentObservation の code 属性。

XPath:

OrderSetHeader/pertinentInformation/pertinentObservation/code

### 1.13.3. 診断コード情報

■定義

病名を表すコード。病名は、コードの表示名として指定する。表示名とは別にコードのオリジナルテキストを設定する必要がある場合には、CodeText を使用する。

ValueSet: ICD10 を使用する。CE 型。

■現行対応項目:

- 病名 CD1-10 (ユヤマ向け電文基本情報部 32-50)
- 病名 1-10 (ユヤマ向け電文基本情報部 33-51)

■インターフェース

**string IDiagnosis.Code [get;]** (HL7 必須属性)

病名コードを object として取得する。

**string IDiagnosis.CodeText [get;]**

コードのオリジナルテキストを取得する。

コードの文字列表現を得たい場合には、

```
string ICD10.Singleton.ToString(object);
```

を使用する。

コードの表示用文字列を得たい場合には、

```
string ICD10.Singleton.DisplayNameOf(object);
```

を使用する。

特定のコードとの等価性を調べる場合には、

```
bool ICD10.Singleton.<Get property for a specific  
code>.Equals(object);
```

を使用する。

#### ■HL7 マッピング(参考)

OrderSetHeader.pertinentInformation.pertinentObservation の value 属性。

XPath:

```
OrderSetHeader/pertinentInformation/pertinentObservation/value
```

#### 1.13.4. 診断日情報

##### ■定義

この病名が付けられた診断日。

##### ■現行対応項目:

- なし

##### ■インターフェース

```
DateTime IDiagnosis.Date [get;]
```

診断日を返す。値がない場合は new DateTime()を返す。

#### ■HL7 マッピング(参考)

OrderSetHeader.pertinentInformation.pertinentObservation.author の time 属性。

XPath:

```
OrderSetHeader/pertinentInformation/pertinentObservation/author/time
```

#### 1.13.5. 診断医師 ID



## ■定義

診断医師を一意に識別する ID。医師 ID と、病院情報システムのユーザ ID の 2 種類の ID を使用可能。

## ■現行対応項目:

- なし

## ■インターフェース

**string IDiagnosis.DiagnosticianId [get;]** (HL7 必須属性)

診断医師 ID(Extension)を取得する。診断医師 ID(Extension)を持たない場合には null を返す。

**string IDiagnosis.DiagnosticianRootUid [get;]** (HL7 必須属性)

診断医師 ID(RootUID)を取得する。診断医師 ID(RootUID)を持たない場合には null を返す。

**string IDiagnosis.DiagnosticianIdIssuingAuthorityName [get;]**

診断医師 ID の ID 発行機関名称を取得する。ID 発行機関名称を持たない場合には null を返す。

**string IDiagnosis.DiagnosticianUserId [get;]** (HL7 必須属性)

診断医師の病院情報システムユーザ ID(Extension)を取得する。ユーザ ID(Extension)を持たない場合には null を返す。

**string IDiagnosis.DiagnosticianUserRootUid [get;]** (HL7 必須属性)

診断医師の病院情報システムユーザ ID(RootUID)を取得する。ユーザ ID(RootUID)を持たない場合には null を返す。

**string IDiagnosis.DiagnosticianUserIdIssuingAuthorityName [get;]**

診断医師の病院情報システムユーザの ID 発行機関名称を取得する。ID 発行機関名称を持たない場合には null を返す。

## ■HL7 マッピング(参考)

OrderSetHeader.pertinentInformation.pertinentObservation. assignedEntity の id 属性。医師 ID を最初の要素に、ユーザーID を 2 番目の要素に入れる。

XPath:

OrderSetHeader/pertinentInformation/pertinentObservation/author/assignedEntity/id

## 1.14. 薬剤師情報

薬剤部モニタでオーダーの送信処理を行なった薬剤師を指定する。  
薬剤師情報は ControlAct に設定する。

### 1.14.1. ControlAct の MoodCode 情報

#### ■定義

ControlAct の MoodCode を指定する。

#### ■現行対応項目:

- 対応項目なし。

#### ■インターフェース

**object IPharmacist.MoodCode [get;]** (HL7 必須属性)

ControlAct の MoodCode。値を持たない場合には null を返す (IControlActBase から継承)。

処方オーダーの状態変更メッセージの場合、常に

Nori.HL7.Version3.VocabularyDomains.ActMood.Singleton.EventOccurrence; (EVN)  
を指定。

### 1.14.2. オーダ送信日時

#### ■定義

薬剤師がオーダーの送信処理を行なった日時。

#### ■現行対応項目:

- 調剤日
- 発行日

#### ■インターフェース

**DateTime IPharmacist.IssuedDateTime [get;]**

薬剤師がオーダーの送信処理を行なった日時を取得する。送信日時を持たない場合には new DateTime() を返す。

### 1.14.3. 薬剤師ユーザ ID

#### ■定義

薬剤部モニタでオーダーの送信処理を行なった薬剤師のユーザ ID。

#### ■現行対応項目:

- 新規追加 調剤指示送信者名ユーザ ID

#### ■インターフェース

**string IPharmacist.UserId [get;]**

薬剤部モニタでオーダーの送信処理を行なった薬剤師のユーザ ID(Extension)を取得する。  
ユーザ ID(Extension)を持たない場合には null を返す。

**string IPharmacist.UserRootUid [get;]**

薬剤部モニタでオーダーの送信処理を行なった薬剤師のユーザ ID(RootUID)を取得する。  
ユーザ ID(RootUID)を持たない場合には null を返す。

**string IPharmacist.UserIdIssuingAuthorityName [get;]**

薬剤部モニタでオーダーの送信処理を行なった薬剤師のユーザ ID 発行機関名称を取得する。  
ID 発行機関名称を持たない場合には null を返す。

### 1.14.4. 薬剤師氏名

#### ■定義

薬剤部モニタでオーダーの送信処理を行なった薬剤師の氏名。漢字氏名とカナ氏名を姓、名に分けて表現する。

#### ■現行対応項目:

- 新規追加 調剤指示送信者名 (漢字)
- 新規追加 調剤指示送信者名 (カナ)

#### ■インターフェース

**string IPharmacist.WholeName [get;]** (メッセージ作成時実装不要)

薬剤部モニタでオーダーの送信処理を行なった薬剤師の姓名(漢字)を取得する。値がない