

```
(void)setSalary:(unsigned int)salary  
(unsigned int)salary
```

For the `setSalary:` method, `takeValuesFromDictionary:` converts the object value for the `^salary^` key in the dictionary to an unsigned int and passes it as salary. Similarly, `valuesForKeys:` converts the return value of the salary method to an NSNumber and inserts that into the dictionary that it returns.

The default implementations support the following scalar types:

```
charunsigned char  
shortunsigned short  
intunsigned int  
longunsigned long  
floatdouble
```

Object values are converted to these types with the standard messages `charValue`, `intValue`, `floatValue`, and so on. Note that the key-value coding methods don't check that an object value actually responds to these messages this can result in a run-time error if the object doesn't respond to the appropriate message.

An important issue to consider in using C scalar types is that most relational databases allow the use of a NULL value distinct from any numeric value, represented in the Enterprise Objects Framework by the `EONull` class. Since the C scalar types can't accommodate a distinct NULL value, the default implementations of the key-value coding methods raise `NSInvalidArgumentException` on encountering an `EONull` object that needs to be converted. You should either design your database not to use NULL values for numeric columns, or design your enterprise object class to use NSNumber objects where NULL values are allowed.

takeValuesFromDictionary: valuesForKeys:
Resetting key bindings flushKeyBindings

(void)flushKeyBindings

Invalidates the cached key-binding information for the receiver's class. The Enterprise Objects Framework caches key-binding information to optimize the default implementations of the other methods of this protocol by caching key-binding information and instance variable type information.

This method should be invoked whenever a class is modified or removed from the runtime system.

(BOOL)takeValuesFromDictionary:(NSDictionary *)aDictionary

Sets properties of the receiver with values from aDictionary. Returns YES if the receiver read all values from aDictionary, NO if it couldn't take all values.

