

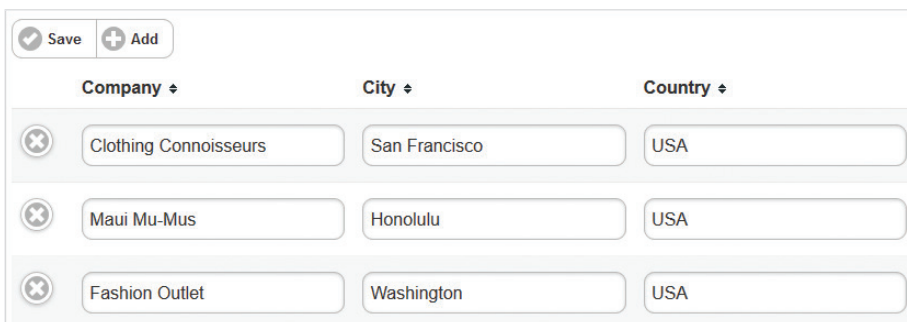
What's New in OpenText™ Gupta TD Mobile 2.1

One team, one IDE, one source – all mobile devices

Developing native mobile workforce apps requires developers for iOS® and Android™ having very specialized and expensive skills. Gupta TD Mobile offers one integrated IDE that uses high-level coding and data-access-configuration to quickly build native mobile workforce apps that work on all mobile devices, regardless of screen size, language, and mobile operating system at a fraction of the cost of native development.

Fully Editable Table Control

Enable your app users to edit data directly in a table cell, without moving to a separate editing screen. Table controls can now insert new rows of data, update data in existing rows, and delete data rows. Table cells can now contain data fields, a flip switch, or a combo box. For developers implementing the delete, insert and update, functionality is easy and can be accomplished using Gupta TD Mobile no-coding Data Operations or by creating self-written back-end code.



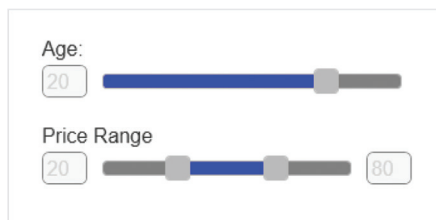
	Company ↕	City ↕	Country ↕
✕	Clothing Connoisseurs	San Francisco	USA
✕	Maui Mu-Mus	Honolulu	USA
✕	Fashion Outlet	Washington	USA

NEW FEATURES

- Integrate data silos, create apps that use REST services to connect to any software service.
- Faster from idea to reality. Higher developer productivity and automated configuration for staging and production environments.
- Improved high usability UX, table mode data edits, and new UX controls increase usability for mobile workforce apps.
- Many other enhancements that increase developer productivity.
- New API calls for feature-rich apps.

New Slider and Range Slider Control

The new slider control gives a nice UI for users to select a numeric value in a specific range. For example, set your max price for filtering cars. Slider control also has an option to turn it into a range slider, where users can select both an upper and lower value, for example, a filter where a user can set the price range of cars they're looking for. The maximum value and minimum value can be set in the control's properties. For range sliders, an upper and lower value bind is available to easily retrieve the user settings.



Conditional Data Formatting

Provide flexible visual color hints for data that is being displayed. For example, show invoice amounts that are late in red, show paid invoice amounts in green, and so on. The new color mapping editor lets developers define multiple color values and lets them define the logic that will be used to evaluate the color at runtime.

Color Mappings Editor ✕

Bind Field
The field value to test against for each color mapping

AccountStatus

Mappings

Add Remove

'{0}' == 'Late'	Red
'{0}' == 'Paid'	Green
'{0}' == 'Pending'	Blue

Expression

JavaScript expression format string, which when true will apply the selected color to the text. Any instance of {0} will be replaced with the field value. Examples: {0} > 5, '{0}' == 'Failed', {0} != 10

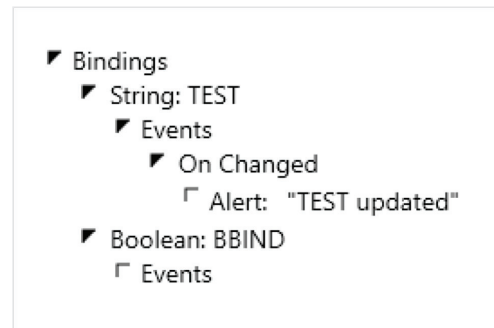
'{0}' == 'Late'

Color

Cancel Save

New Bind Events

Bind events enable developers to add client side code when the value of a binding has changed. When data has changed, this can be used to re-calculate values or to initiate operations automatically.

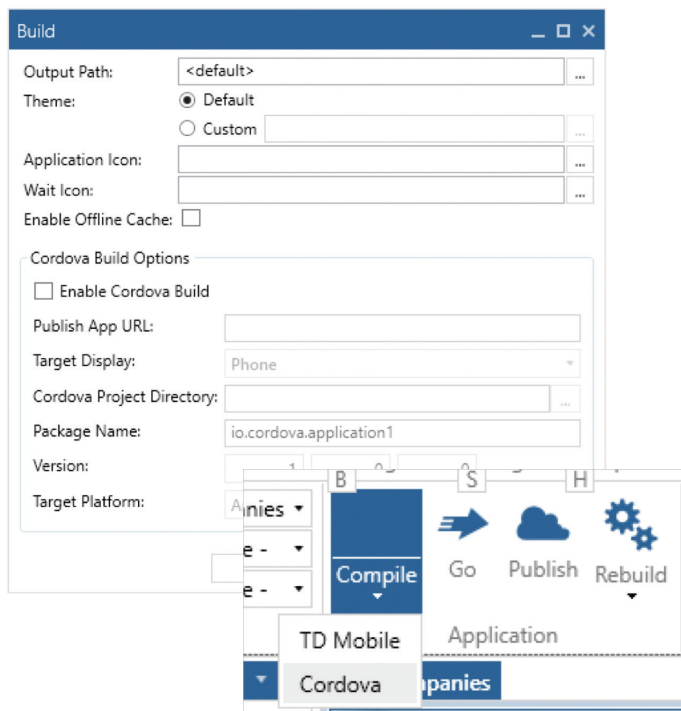


New Page Events

With Gupta TD Mobile 2.1, we are introducing two new page level events, useful for responding to changes in network status and device orientation. On OnlineStatusChange is raised when the devices online/offline status has changed. On OrientationChange is now raised when the device changes from portrait to landscape viewing mode.

Easy Native Compilation

Compiling Gupta TD Mobile applications for native deployment has become much easier and quicker. With Gupta TD Mobile 2.1, developers do not need to step outside of Gupta TD Mobile to start and configure the native app compilation process. New settings in the build dialog let developers configure the native app. A new compile option in the Project Ribbon triggers the native app compilation process.

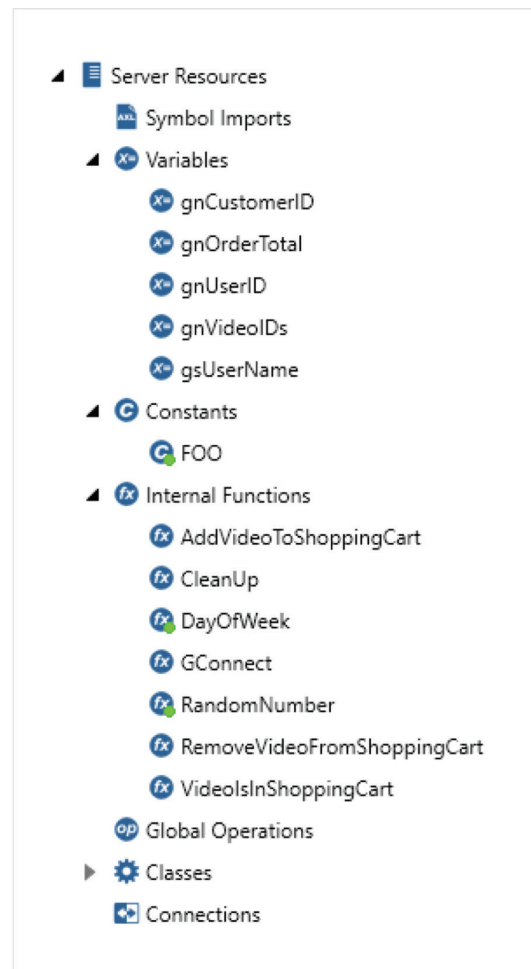


Easy Deployment Back-End Configuration

New back-end configuration options allow developers to create configuration file settings that are being read at runtime to cope for custom database connection strings and any other configuration setting they wish to create. At runtime, Gupta TD Mobile can read the values from the configuration file to adopt to custom deployment environments. Database connection strings can be pre-defined for staging and production sites without the need to modify apps for production environments.

Visual Indicator for Included Objects

The Gupta TD Mobile 2.1 object tree uses a green dot to indicate whether objects are included via a code library.



New Web APIs

Several new APIs have been added to retrieve a client's IP address, the client's app URL for constructing resource links and APIs for retrieving the configuration file settings for database connection strings and user defined configuration variables.