tf.contrib.layers.create_feature_spec_for_parsing

```
create_feature_spec_for_parsing(feature_columns)
```

Defined in `tensorflow/contrib/layers/python/layers/feature_column.py`.

See the guide: Layers (contrib) > Feature columns

Helper that prepares features config from input feature_columns.

The returned feature config can be used as arg 'features' in tf.parse_example.

Typical usage example:

```
# Define features and transformations
feature_a = sparse_column_with_vocabulary_file(...)
feature_b = real_valued_column(...)
feature_c_bucketized = bucketized_column(real_valued_column("feature_c"), ...)
feature_a_x_feature_c = crossed_column(
  columns=[feature_a, feature_c_bucketized], ...)

feature_columns = set(
  [feature_b, feature_c_bucketized, feature_a_x_feature_c])
batch_examples = tf.parse_example(
    serialized=serialized_examples,
    features=create_feature_spec_for_parsing(feature_columns))
```

For the above example, create_feature_spec_for_parsing would return the dict: { "feature_a": parsing_ops.VarLenFeature(tf.string), "feature_b": parsing_ops.FixedLenFeature([1], dtype=tf.float32), "feature_c": parsing_ops.FixedLenFeature([1], dtype=tf.float32) }

Args:

- `feature_columns`: An iterable containing all the feature columns. All items should be instances of classes derived from _FeatureColumn, unless feature_columns is a dict – in which case, this should be true of all values in the dict.

Returns:

A dict mapping feature keys to FixedLenFeature or VarLenFeature values.

Last updated November 2, 2017.