```
// ...initialize matrices and transpose them.

ConstTable->SetMatrixTransposeArray(Device, handle, M, 4);
```

- **SetMatrixTransposePointerArray**: يستخدم لوضع قيم نسق من المؤشرات يشير كل منها إلى منقول مصفوفة 4×4:

```
D3DXMATRIX M[4];

// ...allocate and initialize matrices and transpose them.

ConstTable->SetMatrixTransposePointerArray(Device, handle, M, 4);
```

- **SetVector**: يستخدم لوضع قيم متحول من النوع D3DXVECTOR4:

```
D3DXVECTOR4 v(1.0f, 2.0f, 3.0f, 4.0f);
ConstTable->SetVector(Device, handle, &v);
```

- **SetVectorArray**: يستخدم لوضع قيمة متحول عبارة عن نسق من الأشعة:

```
D3DXVECTOR4 v[3];

// ...initialize vectors

ConstTable->SetVectorArray(Device, handle, v, 3);
```

- **SetValue**: يستخدم لوضع قيمة من نوع ذو حجم غير مبين مثل بنية. مثلاً لاستخدام SetValue لوضع قيمة من النوع D3DXMATRIX نكتب:

```
D3DXMATRIX M(...);
ConstTable->SetValue(Device, handle, (void*)&M, sizeof(M));
```

16.2.1.3: تحديد قيم الثوابت الافتراضية

يقوم المنهج التالي بوضع الثوابت بقيمها الافتراضية، والتي هي القيم التي تم استخدامها في تهيئة الثوابت عندما تم التصريح عنها. يجب استدعاء هذا المنهج مرة واحدة أثناء إعداد التطبيق:

```
HRESULT ID3DXConstantTable::SetDefaults(
    LPDIRECT3DDEVICE9 pDevice
);
```

- **pDevice**: مؤشر إلى الجهاز المخصص لجدول الثوابت.